

Computer Facility User's Guide

April 1985

**Network Information Center
SRI International
333 Ravenswood
Menlo Park, California 94025**

Table of Contents

1. ACTION	5
2. SITE OPTIONS	7
3. INTRODUCTION	11
4. DEFINITIONS	13
5. HELP and DOCUMENTATION	15
6. ACCESSING THROUGH A MILNET OR ARPANET TAC	17
7. ACCESSING THROUGH THE MICOM	21
8. ACCESSING THROUGH A DIAL-UP LINE	23
9. GETTING STARTED	27
10. DIRECTORY INFORMATION	31
11. EXEC	37
11.1. BASIC INFORMATION AND COMMANDS	37
11.2. DIRECTORY-RELATED TOPICS AND COMMANDS	38
11.3. SYSTEM-RELATED COMMANDS	40
12. MESSAGE SYSTEMS	41
13. MM	43
13.1. MM - PART ONE	44
13.1.1. Sending a Message	45
13.1.2. Send Mode Commands	45
13.2. MM - PART TWO	47
14. MAIL-RELATED PROGRAMS	49
15. TEXT EDITING PROGRAMS	51
16. EMACS	53
16.1. EMACS - BASIC COMMANDS	54
16.2. EMACS - BASIC CURSOR COMMANDS	55
16.3. EMACS -- ADDITIONAL FEATURES	56
16.4. EMACS -- LIST OF ALPHABETIC COMMANDS	58
16.5. EMACS -- LIST OF OTHER USEFUL COMMANDS	60
17. FTP -- FILE TRANSFER PROTOCOL	61
18. THE TN PROGRAM	63
19. THE ARCHIVE/MIGRATION SYSTEM	65
19.1. ARCHIVING - OVERVIEW	65
19.2. MIGRATION - OVERVIEW	66
19.3. MIGRATION - POLICY	66
19.4. COMMANDS - DESCRIPTIONS AND EXPLANATIONS	67
19.4.1. GETTING ARCHIVE-STATUS INFORMATION	67
19.4.2. CHECKING ON FILES TO BE MIGRATED	68
19.4.3. CANCELING AN ARCHIVE REQUEST	69
19.4.4. RETRIEVING AN ARCHIVED OR MIGRATED FILE	70
19.4.5. CHECKING THE STATUS OF YOUR RETRIEVAL REQUEST	70

20. SCRIBE -- BASIC COMMANDS	75
20.1. FILENAME TYPES	76
20.2. GLOBAL COMMANDS	77
20.3. USEFUL SCRIBE COMMANDS	79
20.4. SENDING AN ".MSS" FILE THROUGH SCRIBE	81
20.5. EXAMPLE OF .MSS AND .IMP FILE	82
20.6. TABS	84
20.6.1. Setting Tabs	85
20.6.2. Tabs used with the Flushright Command	85
20.6.3. Setting Tabs Visually	85
20.7. USING SCRIBE FOR LETTERS	86
21. USEFUL PROGRAMS	89
22. PROGRAMMING LANGUAGES	91
23. APPENDIX	93
1. TOPS-20 Commands with Brief Descriptions	93
Useful Addresses	97
Suggested Reading	99

List of Figures

Figure 13-1:	Sample message	43
Figure 17-1:	Example of FTP SEND Command Sequence	62
Figure 17-2:	Example of FTP GET Command Sequence	62

PREFACE

This manual is based on *User's Guide to TOPS-20*, written at Information Sciences Institute, 4676 Admiralty Way, Marina del Rey, CA 90292. Revised for the Network Information Center by Stacia Snapp.

-- April, 1985

1. ACTION

Any questions, suggestions, problems or requests for documentation may be addressed to ACTION@SRI-NIC. ACTION's mail file is scanned frequently throughout the day and all requests are acted upon by the operator(s) on duty. In an urgent situation, users may call (415) 859-5921 or (415) 859-4664.

2. SITE OPTIONS

Key:
 ↑ = Control Key
 ⌘ = Escape Key
 ↵ = Carriage Return

This section provides the experienced user with a quick guide to some of the "site options" at the SRI-NIC. These are programs and practices which may be different at other Tops-20 sites.

Archive policy Archive runs are done "as needed." The number of disk pages marked for archival is checked periodically by the Computer Operator. When a significant number of pages have been marked for archival or two weeks have passed since the last archive, an archive run is done. To find out when the last archive was run, type help status↵

Backup schedule

Incremental backups to magnetic tape are done nightly, and full dumps are done every Saturday. Every three months one set of full dumps is set aside as a permanent archive of the System. help perm-dates↵ will return a list of the dates on which permanent dumps were done. The NIC only does archives on the PS: structure, since SS: is intended to be used as a supplementary work area and permanent files should not be kept there.

Consulting

If you need assistance in the use of the System, the first person you should contact is the Computer Operator. More complicated problems can be taken to other facility staff members. Requests for major levels of effort should be directed to the facility manager. Try to balance requests to various staff members rather than always going to the same person.

Default directory parameters

The default and strongly recommended directory protection (774040) permits access to group and non-group members only according to individual file protections. The default file protection (775200) permits read access only to group members and no access to non-group members.

File generation retention count is set to 0 by default which causes all generations of files to be kept.

Default editor The default editor is EMACS. To change this default to another editor (such as the line editor SYS:EDIT), type define editor: sys:EditorName↵

Directory creation procedure

There is an online template to complete when you would like to have a directory set up. It is in DOC:DIRECTORY-REQUEST.TXT. To use it, copy it to your area, edit it and fill out all appropriate sections, and then mail it online to ROODE and VIVIAN. This template should be used for all directory creation requests.

Disk organization

Login directories are located on the PS: structure. Currently there is one other filesystem available for general use, SS:. This secondary structure is subject to inavailability in the event of a severe hardware problem, so all files critical to operation must be located on the PS: structure. Projects involving large amounts of disk space (especially in the form of temporary or "scratch" files) should make use of SS: to avoid overtaxing the space available on PS:.

Manuals

Documentation is issued by the Computer Operator upon request. To request a

document, send mail to ACTION@SRI-NIC. If a manual is not in stock, it will need to be ordered, so a delay is possible. Popular manuals available include the EMACS Reference Guide, the Scribe Document Production System User Manual, and the TOPS-20 Commands Reference Manual. The NIC Reference staff keeps a reference set of manuals that is accessible to any NIC staff member.

Migration policy

Files in PS: directories which have not been touched in 122 days are moved (migrated) to tape on the first day of each month. A warning message is sent through electronic mail to the users involved, specifying which files will be migrated (if they are not touched before migration takes place).

Not-logged-in services

The NIC supports several programs which can be used by not-logged-in users. These include the WHOIS/NICNAM program, which allows access to the information in the NIC's user registration database, and TACNEWS, which provides easy perusal of certain sets of news files and documents available on the NIC.

Operations schedule

The NIC system normally operates 24 hours a day 7 days a week. Problems or questions concerning access to the NIC system may be directed to the Computer Operator at (415) 859-5921 or 859-4664. A Computer Operator is on duty weekdays 7am to 3pm, 4pm to 9pm and 10 pm to 2am time, and weekends 11am to 3am Eastern time.

Since the NIC system is used frequently by people on the East Coast, we make a special effort to keep the System up and running between 4 a.m. and 5 p.m. to provide service for them, as well as NIC staff members, during standard working hours.

Downtime for Preventive Maintenance is scheduled for several hours during the evening of the fourth Thursday of each month. A message is sent to all NIC staff members reminding them of this about a week before the downtime is scheduled will take place. Type info downtime for a list of all scheduled downtimes.

Output

Computer output is produced in room EJ214. The equipment operates on a self service basis, with assistance available from the Computer Operator. The lineprinter should be used where possible instead of the more expensive laser printer, especially where large listings are involved.

Site configuration

The Network Information Center's SRI-NIC machine is a DECSYSTEM-2065 mainframe with 3 megawords (MW) of 36-bit main memory and 261MW of disk, having a power equal to 10,000 butterfly sneezes. The exact machine configuration is as follows:

KL10-E Model B Processor:

- 2 MG20 Memory controllers
- 3 MW MG20 Memory
- MCA25 Cache Memory
- 2 AN20 ArpaNet interfaces
- Network 10 (ARPANET) and 26 (MILNET)
- 3 RH20 Massbus Channels
- DIB20 Adapter for multiplexed I/O Bus

PDP-11/40 Front End Processor:

RX02 Dual floppy disk drives
 3 DH11 Terminal interfaces
 3 @ 16 TTY lines each = 48 lines
 RH11 Massbus Channel
 LP20 Lineprinter interface

Peripherals:

2 RP07 disk drives
 111MW each
 RP06 disk drive
 39MW
 261 MW Total disk storage

 2 TU78 1600/6250-BPI tape drives
 LP27 800 LPM Lineprinter

System-wide logical names

The following logical names of general interest are defined system-wide on SRI-NIC:

BBD:	Electronic bulletin board files
DOC:	System Documentation files
EDITOR:	Default choice of text editor (SYS:EMACS)
EMACS:	Library files for EMACS text editor
FONT:	System utility Raster font files
HLP:	System help files
PCL:	Programmable Command Language library
SAI:	SAIL programming language library
SCRIBE:	Scribe database location

The following logical names are used to implement the phasing in of system utility programs:

PUB:	Current production versions
NEW:	New versions undergoing tests
OLD:	Old versions recently replaced on PUB:
SYS:	Defined to select one of above (defaults to PUB:)

The logical names PUB:, OLD:, and NEW: all default to searching the user's connected directory as the last step.

The following logical names point to data used for the mission of the NIC:

DDN-NEWS:	DDN Newsletters and Management Bulletins
IEN:	Internet Experimental Notes
NETINFO:	Network Information Files
QUERY:	Data available through Query program
RFC:	Requests for Comments (RFC's)

The following logical name is used by a specific user community.

REF:	Library for the Reference group
------	---------------------------------

Terminals

Report terminal problems to the Computer Operator. All moves of terminals and other equipment should be reported to ACTION@SRI-NIC so that the inventory records can be updated.

3. INTRODUCTION

This is an introductory document to acquaint new users with some of the most frequently used features of the TOPS-20 operating system. The document is intended for people who would like to use the computer for electronic mail, text editing, and document preparation, but who may not have experience with TOPS-20 or with programming. It does not include an in-depth explanation of the programs covered, but instead shows how to use the basic commands.

The TOPS-20 timesharing operating system is based on the TENEX operating system with some features from the TOPS-10 operating system and a few entirely new features. It is a very powerful system for program development. If you are interested in writing programs on this system, refer to the chapter titled **Programming Languages** in this document.

With the programs presented here, there is often more than one way to achieve the same results. We introduce the simplest set of commands to carry out a task, the assumption being that the user may proceed to more complex documentation after learning these basic commands.

Information about the programs and subsystems supported by TOPS-20 can be found online in the documentation directory, DOC:. Files in DOC: are catalogued according to type (for example, *.manual*, *.doc*, or *.changes*.)

In this document, the underlined words are what you type to execute commands. Underlined words that are also italicized are not meant to be typed literally, but represent individual names. For example, when you see

Password:*Your Password*

you should type your own password, rather than the words "Your Password." Another word that you will often see (underlined and italicized) is *FileName*. You must always substitute the actual name of the file that you are working with for the word *FileName*. Generally, the System's response to your typed command is not shown, but only each sequence that you must type.

Note that in this document you will find repeated references to the **Carriage Return**, the **Control Key**, and the **Escape Key**. Since they are of such importance to the user, they have been assigned specific symbols, defined on the following page.

The **Carriage Return** is represented in this document by $\Leftarrow$. It confirms a given command and most commands are not executed until you type a carriage return. If you type a command and the cursor sits blinking after the last character has been typed, you probably need to type a carriage return.

The **Control Key** is represented in this document by $\uparrow$. It is used with another key to enter a command. You may hold the Control Key down for as long as you want, but you should type the other key only once (the way you would use the Shift Key on a typewriter).

The **Escape Key** is represented in this document by $\simeq$. This key usually completes a command or a FileName. As soon as you have typed enough of a command or a FileName for the System to recognize it, type the Escape Key to complete it. If you haven't typed enough, the System simply beeps and waits for you to type more. At the Exec level¹ it is not necessary to use the Escape Key on commands as long as you type a space instead. When you use the Escape key, though, you are prompted for the next part of the command. The System does require that DirectoryNames and FileNames be typed completely before it recognizes them, so the Escape Key saves you the effort of typing out the complete name yourself.

¹See next chapter for definition of Exec.

4. DEFINITIONS

Key:
 ↑ = Control Key
 ⌘ = Escape Key
 ↵ = Carriage Return

Cursor Cursor styles vary from terminal to terminal. A cursor indicates where the next character you type will appear on your screen.

Default Whenever the System does something that is **assumed**, rather than something you've specified, that is the Default. For example, if you give the command to see the contents of a file by typing Type FileName ↵, the System will assume that the file is in your Directory. That is the Default. If you want to see a file in another Directory, you must override the default by specifying the Directory in **angle brackets** before the FileName. When you want something other than the System Default, you must type more information.

Directory A Directory is a collection of files. Your Directory is your working space on the System. You can think of it as your office with your files in it. When you login (identify yourself to the System by giving your DirectoryName and Password), you are given access to your Directory and files. Whenever you see a name enclosed by angle brackets like this

<Name>

or with angle brackets preceded by a structure name like this

PS:<Name>

that name is a DirectoryName.

File A file is a specific collection of data such as a letter, a list, or a report. Files have names to enable the computer to recognize them. A file can be long (an entire document) or short (one line).

Logical Names

Logical names are a shorthand means of specifying a directory, a filename, or part of a filename. They consist of a string followed by a : (colon). Some logical names are set up by the System and can be used by everyone. You may also create your own logical names, which you can then use during your current session. If you put a list of logical names which you commonly use in your Login.Cmd file, those logical names will be available for you whenever you login.

Type help logical-names↵ for more information about how to set up and use logical names.

FileNames

A FileName is the name of a file and is structured in a specific way. A FileName is composed of three parts: Name.Type.Generation. Each part of the FileName is separated by a period. **No spaces are permitted in FileNames.** When you name a file, it is important to choose a name that is not only unique (so you don't have to type the entire name before typing the Escape Key) but one which will also remind you of what is in the file. It may be any combination of up to 39 alphabetic and/or numeric characters. Quite often, a FileName is longer than one word. Since you may not use spaces to separate the words, you may use hyphens. Sometimes, though, the FileName Type is significant and indicates an important aspect of the file. This is particularly true with Scribe (a formatting program) files. Generations are

automatically given to files by the System. Each time a file is updated, it is given a higher number, so the most recent version of a file has the highest Generation number.

Password Your Password is the key to your Directory. When you login, you type your **DirectoryName** and then your Password. When you type your Password, it will not appear on the terminal screen. This allows you to keep your Password a secret.

Program Programs issue instructions to the computer. Many programs process files. For example a **message program** such as MM can create and send a message and manipulate a file of messages. A **text editing program** such as EDIT or EMACS can follow commands to create a text file and edit it, while a **text formatting program** like Scribe can follow commands to format text by centering a heading, enumerating a list, italicizing a name, etc. Programs are stored as files on the System.

Structure

Structures are another way of categorizing information kept on the System. Generally structures correspond to the physical disks on which files are kept. Structure names are usually only a few characters long and end with a : (colon). The three structures on SRI-NIC are PS:, SRC:, and SS:.

Each directory on the system has a structure associated with it. The full name for a directory includes its structure name, like this:

PS:<DirectoryName>

Directories which can be logged into are always on the PS: structure. Thus when you login, the directories you specify without a structure name will by default use the PS: structure. If you need to do something with a directory which is on another structure, you will need to specify that structure whenever you use the directory name. For example:

@type src:<DirectoryName>photo.log

TOPS-20 (EXEC)

TOPS-20² is the main program or operating system of the computer. It interprets all of the commands and programs that you use. It is commonly referred to as the **System**. Logging in to the system places you at the EXEC level of TOPS-20. You must also log out from the EXEC. Although it is referred to as a "level," the EXEC is really just another program that processes all commands by making system calls. From this "level" you enter different programs, and when you leave a program you return to it. When you are at the EXEC, you should always see the "@" sign on the left side of the screen. This is the EXEC "prompt."

²TOPS-20 is a product of Digital Equipment Corporation.

5. HELP and DOCUMENTATION

Key:
↑ = Control Key
⌘ = Escape Key
↵ = Carriage Return

TOPS-20 has a valuable command called **HELP**³ which will print documentation on various system commands and programs. If the user types

@help ?

the EXEC will print a list of commands and features for which further documentation is available. To obtain further assistance on a specific command, the user should type

@help command↵

For example, to see detailed information about the "Delete" command, you would type

@help delete↵

³At the back of this manual you will find an appendix which lists TOPS-20 commands and a brief description of their functions.

A directory called PS:<DOCUMENTATION> exists on each TOPS-20 system and is accessible to each logged-in user. Files in this directory contain detailed information on a variety of subjects and available programs. These files have the *subject* (Pascal, Scribe, etc.) in the FileName field, and "doc" in the FileType field.

The logical name for this directory is DOC:. Since this logical name may indicate other directories which hold documentation in addition to PS:<DOCUMENTATION>, you will have the best chance of finding documentation files if you use the logical name instead of the directory name.

For example, to see the current on-line information on Pascal, you would type

@type doc:pascal.doc←

Revisions and changes are stored in files which have "*changes*" in the FileType field. These *.changes* files are in a message format. To see changes in Pascal programs, type

@type doc:pascal.changes←

A list of current subjects and programs for which information is available will be listed by typing

@dir doc:*.*.*←

6. ACCESSING THROUGH A MILNET OR ARPANET TAC

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

Baud Rate

The baud rate determines the speed at which your terminal receives input and produces characters on its screen. Generally speaking, the higher the baud rate setting, the faster the display. (Other factors, such as the load average of the system, may influence this as well.) If you are on a TAC, perform the following sequence so that the TAC will communicate with the terminal at a proper baud rate:

Type the BREAK key
 Type iq (control-Q)

The TAC will print "<sitename>TAC<version #><port #>" when the proper baud rate has been discovered.

Giving TAC Commands

All TAC commands begin with an intercept character ("@"). The command text follows in the form of words separated by spaces. Only the first character of each word is recognized; in fact, other characters are ignored, so one may type abbreviated commands (see "Command Styles" section in the next chapter). Upper and lower case letters may be freely intermixed. Some commands must be completed by entering a decimal number. All commands must be confirmed by a carriage return.

Opening a Connection to a Host

The user should type

@ open address⇐

When going through an ARPANET TAC, the user will then be greeted with the banner of the system accessed. The official ARPANET address for SRI-NIC is 10.0.0.51, but that may be abbreviated when using the TAC to 0/51 (or simply 51). **Note that users coming in through an ARPANET TAC will be able to access ARPANET systems only.**

When using a MILNET TAC, the user needs to list his or her TAC UserID and Access Code before the banner of the system being accessed will appear.

On the following page is an example of the login dialog for a user trying to access SRI-NIC through a MILNET TAC. The user input is underlined.

- a) PVC TAC 110 #:01
- b) @o 26.0.0.73←
- c) TAC UserID: User's ID←
- d) Access Code: User's Access Code← (Does not echo.)
- e) Login OK
- f) TCP trying...Open

In this example, the TAC prints its greeting (a). The user proceeds to give the command to open a connection to host 26.0.0.73 (b). If you are used to using a host number with a slash in it, you may still type that form, i.e., @o 0/73, as long as you are not going across networks. At the TAC UserID prompt, the user enters his or her UserID (c). The Access Code corresponding to the UserID is entered (d). After a brief wait, the TAC indicates that login is successful (e) and goes on to the normal connection sequence (f).

When you are entering your TAC UserID and Access Code:

- A carriage return (indicated as ← in the example) terminates each input line and causes the next prompt to appear.
- As you type in your TAC UserID and Access Code, it does not matter whether you enter an alphabetic character in upper or lower case. In the typing of the TAC UserID, all lower case alphabetic characters echo as upper case.
- The Access Code is not echoed in full-duplex mode. An effort is made to obscure the Access Code printed on hardcopy terminals in half-duplex mode.
- As an aid to correct reading of Access Codes, they have been designed so that they never contain a zero, a one, a "Q" or a "Z", because each of these characters resembles another. So if you think you see one of these characters in your Access Code, you know it is really the letter "O" [oh], the letter "L" [el], or the number "2" [two].
- You may edit what you type by using the backspace (↑H) key to delete a single character.
- You may delete the entire line and restart it by typing ↑U. A new prompt will appear.
- While entering either the TAC UserID or Access Code, you may type ↑C to abort the login process and return to the TAC command mode. You must interrupt or complete the login process in order to issue any TAC command.

After both the TAC UserID and Access Code are typed into the terminal, the TAC must verify the login attempt. Often this takes less than a second, but in some cases a slight delay will occur. Anything typed at the terminal after the concluding carriage return of the access code, but before the login confirmation (or denial) is received, will generate the message "Wait" on the terminal. If the login is allowed, you will see the message "Login OK" and immediately afterward the TAC will attempt to open your TCP connection. If the login is denied for any reason, you will see the message "Bad Login." The TAC will then prompt you for another UserID and Access Code. After several bad login attempts, the TAC will attempt to hang up your TAC port.

Once logged in, your port remains logged in as long as you have an open connection. There is a ten minute period after you close one connection in which you may open another one without having to go through the login sequence. When you are finished using your TAC port you should log out by typing @logout. Typing @reset has no effect on your login state. If ten minutes go by during which your port does not have an established TCP connection, the TAC will attempt to hang up on your port, logging you out.

To make a connection to SRI-NIC through a MILNET TAC, use 0/73 (or simply 73) or the official network number, 26.0.0.73.

Closing a Connection

When using either MILNET or ARPANET TACS, the user should close the connection after she or he is finished by typing

@logout←

TAC resources are limited, so please remember to log yourself out when you no longer need a connection.

7. ACCESSING THROUGH THE MICOM

Key:
 ↑ = Control Key
 ⌘ = Escape Key
 ↵ = Carriage Return

MAKING A SYSTEM CONNECTION:

1. To make a system connection through the Micom, type a ↵

You will see:

```
Welcome to TSC's Micom.
ENTER CLASS
```

2. Type the system name. To reach SRI-NIC, type NIC↵. You can also reach a number of other systems through the Micom. For example, to make a connection to TSC's Vax 11/750, type TSC↵. For a complete list of the systems to which you can connect through the Micom, type help roadmap↵

If you do not respond to the **ENTER CLASS** prompt within a few seconds, you will see the message:

```
DISCONNECTED
```

Type another ↵ to repeat the **ENTER CLASS** prompt.

3. The Micom will respond with the word **GO**, pause, and then print the system herald (a short block of text identifying the system and its current load average).
4. Now login as you normally would:

```
On TOPS-20 systems (such as NIC):
@Log⌘DirectoryName⌘Password↵
```

```
On Unix systems (such as TSC):
login: DirectoryName↵
Password: password↵
```

ABNORMAL CONDITIONS:

The message **BUSY WAIT?** will appear if all of the lines to the Micom at your terminal's current baud rate are in use. Typing Y↵ will put you in the queue for the next available line. Typing N↵ will repeat the **ENTER CLASS** prompt and you can either change your baud rate and attempt to login to the same system at your new speed, or leave your baud rate unchanged and try to connect to another system.

The message **UNAVAILABLE** may appear if the system is down. If you want to connect to another system, type a carriage return to repeat the **ENTER CLASS** message.

Do not change your terminal speed after you are logged in to a system. If you want to change your speed, use the break key to talk to the Micom. When you see the **ENTER CLASS** message, change your terminal speed and type one or two carriage returns. You should then see the [ENTER CLASS] message again. Now type the system name and login as you normally would.

If you type the **BREAK** key after you are logged in, your job will be detached.

If you see the message **WRONG SPEED** followed by **DISCONNECTED**, the system is not ready for logins. Either wait for that system or make a connection to another system.

If you specify an undefined system, you will be disconnected.

If you have problems at any point in the connection process, type the **BREAK** key on your terminal which will give you the **DISCONNECTED** prompt, then repeat the process from step one.

8. ACCESSING THROUGH A DIAL-UP LINE

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

MAKING A SYSTEM CONNECTION:

There are three methods of accessing SRI-NIC via telephone: direct dialup lines, the Micom dialup, and the ISO dataswitch dialup. A brief description of each follows. See the file HLP:DIAL-IN.HLP for more detailed information.

1. To make a system connection with a direct dialup phone line, dial 328-8731. After the connection has been established, type a ⇐. The system will respond by printing out the system herald followed by the Exec prompt (@).

Now login as you normally would:

@Log≈DirectoryName≈Password⇐

2. If you do not login within 5 minutes, the System will automatically kill the job and your connection will be lost. If this happens, dialup and try again.
3. You can also access SRI-NIC through a TSC Micom dialup line. To do this, dial one of the numbers listed at the end of this chapter under TSC Micom Dialup Numbers. After a connection has been established, type a ⇐.

You will see:

Welcome to TSC's Micom
 ENTER CLASS

To reach the NIC computer, type nic ⇐ to the ENTER CLASS prompt. The Micom will respond, after a pause, with the word GO.

ENTER CLASSnic ⇐
 GO⇐

Slowly type ⇐ until you see the Exec's @ prompt. Now login as you normally would:

@Log≈DirectoryName≈Password⇐

4. The third way you can access SRI-NIC by phone is through the SRI-wide Micom dataswitch. To do so, dial one of the numbers listed under ISO Dataswitch Dial-up Numbers at the end of this chapter. When you have established a connection, you should see the following:

Welcome to ISO's Micom.
 Terminal Problems call X5559.
 Programming questions X4000
 ENTER CLASS

You then type tsc ⇐ to the ENTER CLASS prompt. The Micom will respond, after a pause, with the prompt for the TSC Micom. Now you can proceed as shown above (item 2) when accessing the TSC Micom:

```
Welcome to TSC's Micom
ENTER CLASSnic ←
GO←
```

Once you see the system herald and the Exec prompt (@), login as you normally would:

```
@Log≈DirectoryName≈Password←
```

When accessing the NIC computer through a Micom dialup line, if you do not respond to the ENTER CLASS prompt within a few seconds, you will see the message:

```
DISCONNECTED
```

On a dial-up line, this message will break your connection so you must dial the number again.

ABNORMAL CONDITIONS WITH MICOM CONNECTIONS:

The message **BUSY WAIT?** may appear, if all of the lines to the Micom are in use. Typing Y← will put you in the queue for the next available line. Typing N← will break the connection so you must dial the number again.

The message **UNAVAILABLE** may appear, if the system is down. Dial the number again and select a different system at the **ENTER CLASS** prompt.

Do not change your terminal speed after you are logged in to a system.

If you type the **BREAK** key after you are logged in, your dial-up connection will be disconnected and your job will be detached.

If you see the message **"WRONG SPEED"** followed by **"DISCONNECTED"** the system is not ready for logins. Either wait for that system or make a connection to another system.

If you specify an undefined system, you will be disconnected.

If you have problems at any point in the connection process, type the **BREAK** key on your terminal which will give you the **DISCONNECTED** prompt, then repeat the process from step one.

Please note: the above conditions refer only to Micom dial-up connections, not direct dial-up connections.

TSC Micom Dial-up Numbers

326-4495 (300 or 1200 baud)
326-9246 (300 or 1200 baud)
326-9209 (300 or 1200 baud)
326-4540 (300 baud only)

ISO Dataswitch Dial-up Numbers

859-6279 (1200 baud)
859-6255 (300 baud)

FEX Dial-up Numbers (After Hours)

	300 baud	1200 baud
ISO Micom:		
Sunnyvale (FEX)	(408) 245-6412	(408) 245-6418
San Mateo (FEX)	(415) 349-7597	(415) 349-7658
Oakland (FEX)		(415) 834-7838

9. GETTING STARTED

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

Before logging in

Before you login, make sure that your terminal is plugged in and turned on. Establish a connection to the NIC computer via a direct line, dial-up line, or Micom line. If you are in your office, you will either have your own "direct line" to the computer, or be connected to the TSC Micom (see the chapter titled Accessing Through The Micom information about using the TSC Micom).

If you have your own direct line, begin your terminal session by typing ⇐. The System will respond with the system herald and the Exec's @ prompt.

Logging in

Once your connection has been opened and the host system has provided you with pertinent information about its current status (called the "system herald"), you will see a "@" symbol on the left margin of the terminal screen. This is where you should login by typing your DirectoryName, followed by a space and then your password, i.e.:

@login DirectoryName password⇐

After accepting your DirectoryName and password, the system will display your job number, your terminal number, your last login and current login dates and times, and whether or not any other jobs are currently logged in under your DirectoryName. You will then see another @ at the left of your screen just under all of the other information. This means that the System is ready to accept other commands from you (such as **HELP**) or **FINGER**).

Logging out

The **logout** command ends your current job, deletes all temporary files created during the current logged-in session, expunges deleted files in your directory, checks disk storage allocation, and prints a logout message. You may also log out another job as long as it is logged in under your user name by typing logout JobNumber. You cannot log out another user's job. Use the **logout** command to leave the system overnight.

If you exceed your permanent disk storage allocation⁴ the system prints a message when you log out telling you the number of pages by which you are over permanent storage allocation. Since exceeding your permanent disk allocation can cause problems, a good standard practice is to check your status before logging out by typing information disk-usage⇐ or simply i dis⇐. The system will respond by printing the number of permanent disk pages your directory now occupies. If you are running low on space you can delete unnecessary files at this time.

Detaching a Job

When you login, the System assigns you a job number at the terminal you are using. When you detach, the terminal is released but the job number is still associated with you. When you want to work on the System again, you should **attach** to that job

⁴See the "Disk Allocation" section in the "Directory Information" chapter for more details.

rather than logging in a second job. The detach command is useful when changing terminals or for leaving the System temporarily. To detach your job, type detach←

Attaching to a Job

To return to your detached job, type

att ≈ DirectoryName←
Password←

If you have a detached job at the time you try to log in a new job, the System will ask you if you would rather just attach your detached job. If you would like to do so, respond by typing a ≈ (Do not type any other characters (such as "yes") before typing ≈ or your other job will be left detached). In the first example below, the user confirms to have her detached job attached. In the second example, she chooses to login a new job and leave the first job detached.

Example of attaching detached job while logging in:

```
SRI-NIC, TOPS-20 Monitor 5.3(5751)-1
***
* For TACNEWS, enter: tacnews<RETURN>
* To find the host administrator for host xy-z, enter: whois xy-z<RETURN>
* Report system problems to Action@SRI-NIC or call (415) 859-5921
***
There are 22+7 jobs with load average 1.89
@login (User) stacia (Password) ≈
Job 1 on TTY107 10-Apr-85 11:52am
Last login was Wed 10-Apr-85 11:52am
[Job 23 (Det, ECHO) also logged in under STACIA]
Attach your detached job? [Confirm with carriage return] ≈
Attaching...
@
```

Example of choosing not to attach detached job while logging in:

```
SRI-NIC, TOPS-20 Monitor 5.3(5751)-1
***
* For TACNEWS, enter: tacnews<RETURN>
* To find the host administrator for host xy-z, enter: whois xy-z<RETURN>
* Report system problems to Action@SRI-NIC or call (415) 859-5921
***
There are 20+7 jobs with load average 1.95
@login ≈ (User) stacia ≈ (Password) ≈
Job 1 on TTY107 10-Apr-85 11:53am
Last login was Wed 10-Apr-85 11:52am
[Jobs 23 (Det, ECHO), 24 also logged in under STACIA]
Attach your detached job? [Confirm with carriage return] no ≈
Left detached
@
```


Login Command File

You may create a **login command file** that is read by the System every time you log in. Note: This file must be named **LOGIN.CMD** in order to be read by the System.

If you will be using the same terminal regularly, creating a login command file may be beneficial. In it you can store information such as the type of terminal you use, and the length and width of the "page" (how many lines your terminal screen contains and how many characters each of those lines will hold).

There may be occasions where you wish to change the parameters of your terminal so that they are different from those defined in your login command file. Type **INFORMATION TERMINAL** at the EXEC command level to check your terminal parameters, and if they are not exactly what you would like them to be, use the **TERMINAL ?** command to see what options you have. Then, set the parameters manually by typing at the EXEC level a command such as **ter datamedia-2500** or **ter heath-19**. (Defining a terminal by its *type*, as in the two previous examples, is usually all that is required.) This will override the parameters contained in your login command file, and will automatically set control of output to your terminal.

Take Your login command file is read automatically each time you log in. The System does **not** read this file automatically when you attach, however. You can cause the System to read your login command file by typing

take login.cmd←

Command Styles

Commands can be typed using three styles of input:

Full Input, in which the *complete* command name, argument, and subcommand is typed.

Example: **Information Disk-Usage**←

Recognition Input, in which you type a portion of the command, then type the Escape Key. The System *recognizes* the command and completes typing it for you on the screen.

Example: **i nformation (about) dis k-Usage**←

If you do not type enough of the command for the System to recognize it as a unique, you will hear a bell when you type the Escape Key, instead of seeing the command completed. This means that you will need to type more of the command in order for the System to know which command you want.

Abbreviated Input, in which the unique portion of the command name, argument, and subcommand is typed, without typing the Escape Key.

Example: **i dis**←

10. DIRECTORY INFORMATION

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

Disk Allocation On TOPS-20 each user has two disk space allocations, both of which are strictly enforced. The WORKING STORAGE is the maximum amount of disk you can use while being logged in. PERMANENT STORAGE is the maximum disk space that you are allowed at logout time. TOPS-20 isn't particularly gracious about telling you why you can't work any more once you have actually run out of space. When a system message tells you that you are over allocation, delete and expunge whatever files you don't need. Other files which you might need later on can be marked for archiving or migration. (See the "Archive/Migration System" chapter for detailed information.) For increased disk space allocation, send a message to ACTION.

Generation Retention

The generation number of a file specifies how many times that particular file has been changed. Generation numbers allow the user to create multiple disk files with the same combination of file name and type. The intent of generation numbers is to keep track of successive, modified versions of a single file. You can check your directory's default generation retention number by typing

```
@Information Directory DirectoryName.⇐
@@verbose⇐⇐
```

If your generation retention default is set to 4, for example, it means your directory will retain only 4 generations of any file; the system will automatically delete the older generations in excess of 4. The higher the number, the more versions of the same file will remain in your directory. However, setting the generation retention to 0 allows unlimited generations. The command to change the default is

```
@Set Dir≈ Gen≈ Number⇐
```

This will cause new files which are created in your directory to have the new generation retention count. However, if you wish existing files in your directory to have a certain generation retention count, you must explicitly set that for them with the Set File Generation-Retention-Count *FileNames Number* command. For example,

```
@Set File Gen≈ FileNames Number⇐
```

DIRECTORY Command

The *directory* command is used to find the names of a group of files. For example, typing directory⇐ to the EXEC will print a list of all visible files in your directory.

The *directory* command may be followed by file group specifications, separated by commas, to list the files in the specified groups. If you only want one group, say all the files in another user's directory, you do not need any commas. For example,

```
@directory <roode>⇐
```

If you want to see a list of a particular file type, you can substitute an asterisk (*) for part of the filename(s). The asterisk is referred to as a "wild card." For example:

```
@directory t*←←
```

Typing this command will print a list of all files in your connected directory which begin with the letter "t". Another example:

```
@directory <system>*.mac,<nicprog>*.mac,<unsupported>←←
```

This command will print a list of all files in the <SYSTEM> and <NICPROG> directories which have an extension of .mac, and also a list of all files in the <UNSUPPORTED> directory, since no specification was included after the directory name.

The use of a wild card is a valuable short cut when you want to search a directory only for a particular type of file.

Additionally, there are several convenient variations of the *directory* command that provide certain options. These are:

vdirectory - Lists the names of files, protection, size, the date and time they were last changed, and the writer (no headings).

fdirectory - Lists all the information on files in a format with compressed output without headings.

tdirectory - Lists the names and write dates of files in the order of the date and time they were last changed.

wdirectory - Lists the names of the files, the write dates as well as the writer, and the creator of the file.

qdirectory - Gives information about deleted files.

rdirectory - Lists file names together with the date each file was last read. Files are in order of most recent reading.

DIRECTORY Subcommands

Other options are available with the *directory* command, known as subcommands. You can enter the subcommand mode by typing a comma after the directory command. Your prompt will then be a double EXEC prompt ('@@'). Please note that two carriage returns (indicated in this document by the symbol '←←') are required in subcommand mode to produce the desired output. Here is a list of available DIRECTORY Subcommands and a brief explanation of their functions. You can reproduce this list at your terminal by typing:

```
@directory,← (don't omit the comma)
```

```
@@? One of the following:
```

```
@@ALPHABETICALLY←←
```

Lists files alphabetically. This is also the default of the directory← command to the EXEC, and produces identical output.

```
@@ARCHIVE←←
```

Shows any file with archive status online or offline, visible or invisible.

```
@@BEFORE date←←
```

Shows files written before a specified date.

@@CHRONOLOGICAL $\simeq$ (By) ? one of the following:
CREATION (default), READ, TAPE-WRITE, or WRITE $\Leftarrow\Leftarrow$
 Lists files chronologically.

@@CRAM $\Leftarrow\Leftarrow$
 Omits all columnating spaces between fields.

@@DATES $\simeq$ (Of) ? one of the following:
CREATION, READ, TAPE-WRITE, or WRITE (default) $\Leftarrow\Leftarrow$
 Shows dates for each file

@@DELETED $\Leftarrow\Leftarrow$
 Provides list of deleted files.

@@DOUBLESPEACE $\Leftarrow\Leftarrow$
 Produces alphabetical list of filenames with a blank line separating each.

@@EVERYTHING $\Leftarrow\Leftarrow$
 Lists files alphabetically, and includes detailed information about each file, including its size in disk pages, the size of the file's bytes, the generation retention count, the creation date, last write date, last read date, the tape-write date (if any), the creator of the file, and the last writer to the file.

@@FIND number $\Leftarrow\Leftarrow$
 Lists files with more than a specified number of generations.

@@GENERATION-RETENTION-COUNT $\Leftarrow\Leftarrow$
 Shows the retention count for each file.

@@HEADING $\Leftarrow\Leftarrow$
 Lists files alphabetically (identical output to the directory $\Leftarrow$ command at the EXEC.)

@@INVISIBLE $\Leftarrow\Leftarrow$
 Lists invisible files only.

@@LARGER $\simeq$ (Than) number $\Leftarrow\Leftarrow$
 Lists files larger than a specified decimal number of pages.

@@LENGTH $\Leftarrow\Leftarrow$
 Shows the byte length and size for each file.

@@LPT $\Leftarrow\Leftarrow$
 Sends an alphabetical listing of files to the printer.

@@MIGRATED $\Leftarrow\Leftarrow$
 Lists offline migrated files.

@@NO $\simeq$? one of the following:
 (gives a list of all subcommands for which an opposite is available)
 Negates the given subcommand.

@@OFFLINE $\Leftarrow\Leftarrow$
 Lists visible offline files only.

@@ONLINE $\Leftarrow\Leftarrow$
 Lists only online files.

@@OUTPUT $\simeq$ (TO FILE) filename $\Leftarrow\Leftarrow$
 Outputs an alphabetical list of filenames into specified file.

@@PROHIBIT $\simeq\Leftarrow\Leftarrow$
 Shows files that are set to prohibit migration.

@@PROTECTION==

Gives the protection for each file.

@@REFERENCE ? one of the following:

READ or **WRITE** $\simeq$ ==

Lists files in order of how many times they have been referenced (read or written, depending on the argument given)

@@RESIST $\simeq$ ==

Lists files that are set to resist migration.

@@REVERSE==

Lists files in reverse alphabetical order.

@@SEPARATE==

Provides alphabetical listing of files with a separate line for each filename. This means that if you have four versions of the same file, each version will be listed on a separate line.

@@SINCE *date*==

Lists all files written since a specified date.

@@SIZE==

Shows the page size of each file.

@@SMALLER $\simeq$ (THAN) *number*==

Lists files smaller than a specified decimal number of pages.

@@TIMES $\simeq$ (AND DATES OF) ? one of the following:

CREATION, **READ**, **TAPE-WRITE**, or **WRITE** (default)==

Shows the times for each file.

@@USER $\simeq$ (WHO LAST) ? one of the following:

CREATED or **WROTE**==

Shows either the creator of or the last writer to each file.

File Protection

The protection for each file in your directory consists of a six-digit number, which can be separated into three groups of two digits each. The first two digits represent **owner** access to the file, the second two digits represent **group** access, and the last two digits represent **world** access. By setting the protection for files in your directory, you can control who can and cannot access them.

The default protection for files on the NIC 2060 is 775200. The number 77 in any given field allows full access. This default protection, then, allows the **owner** full access, members of his or her **group** reading, executing, and directory listing privileges, but denies all other users any access at all.

FILE PROTECTION NUMBER

owner	group	world
77	77	00
full	full	no
access	access	access

You can determine the desired protection for a file by adding together the different digits:

FILE PROTECTION DIGITS

02 Permits listing filename with DIRECTORY command
04 Permits appending to the file
10 Permits executing the file
20 Permits changing and/or deleting the file
40 Permits reading the file

For example, as mentioned above, the default file protection number is 775200. This number allows the owner full privileges; members of his or her group are allowed reading, executing, and directory listing privileges (obtained by adding 02, 10, and 40); and all other users are allowed only directory listing privileges. You can set the protection for any file in your directory by typing at the EXEC:

@set file protection filename protection number←

For a more detailed description of file protection and how it works, type help protection← at the EXEC.

11. EXEC

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇑ = Carriage Return

11.1. BASIC INFORMATION AND COMMANDS

- @ The @ is the prompt for the EXEC level. In order to give any commands at the EXEC level you must have the @ prompt.
- Del Key The Del Key deletes characters from right to left. You may need the Shift Key simultaneously. On some terminals the Del Key is labeled the RUBOUT key.
- ↑C You may type ↑C to abort a partially typed or partially completed command (although sometimes you must type several ↑Cs). If you are in the middle of a program, typing a ↑C will return you to the EXEC level. However, this is frequently *not* the best way to leave a program.

The following commands work at the Exec level and in most programs described in this document, except EMACS.

- ↑F This works like the Escape Key on FileNames, but completes only one field of the FileName at a time.
- ↑S To hold the printout on your screen so you can read it, type a ↑S.
- ↑Q Restarts terminal printout that was suspended by ↑S. It will also display the next screenful of text when reading a file.
- ↑R ↑R Retypes the current input line and leaves the cursor positioned at the end of the line.
- ↑T Use ↑T to see the current load average of the System (how busy the computer is). When the load average is high (above 10), the System will respond more slowly to the commands you give. If there is no response at all when you type ↑T, the System may not be working (often referred to as a System "crash").
- ↑U ↑U deletes the entire line from the cursor to the left side of the screen.
- ? To see all of the commands available at the EXEC level, type ?

11.2. DIRECTORY-RELATED TOPICS AND COMMANDS

Key:
 ↑ = Control Key
 ≅ = Escape Key
 ⇐ = Carriage Return

Build Creates, changes, or deletes a directory that is subordinate to your logged-in directory (referred to as a "subdirectory"). Type help subdirectory⇐ for details.

Change Password

See the **Set Directory Password** command in this section.

Copy To copy a file from another Directory to your logged-in Directory (on the same computer):

Cop≅ <DirectoryName>FileName≅⇐

To copy a file from your logged-in directory to another directory:

Cop≅ FileName≅ <DirectoryName>≅⇐

Connect To connect to another Directory:

Conn DirectoryName⇐ Password⇐.

To connect back to your own directory, just type cd⇐ or

Conn⇐

Delete Deletes specified generations of individual files. Type

Delete FileName ≅ ⇐

to delete the lowest (oldest) generation. Deleted files may be retrieved (with the Undelete command) until you Logout or Expunge (or the System crashes).

Directory To see the names of all the files in your Directory, type

Directory⇐.

Directory Subcommands

To get information about a file (for example, size, time of creation, protection status), type

Di≅ FileName ≅,⇐

You will then have a @@ prompt and you can type a !⇐ for a list of available commands.

Expunge Permanently removes *all* deleted files. Type

Expunge⇐

It isn't necessary to specify a *FileName* because the Expunge command works on all deleted files.

FDirectory (Full Directory) Lists the names of files, protection, size, the date and time they were last changed, with no headings. The output is compressed.

Laser To send a file to the laser printer: Laser≅ FileName≅⇐

To cancel a job that you've sent: cancel laser req# ← (find out the req # by typing i o←).

To see other options you may use with the Laser command, type Laser←
FileName←/? or type help laser←

- Password** See the **Set Directory Password** command in this section.
- Print** This command, followed by a FileName, will produce a hard copy of that file on the NIC lineprinter. Type
print FileName←
- Protection** On the NIC TOPS-20 system, the protection default parameters allow the System and the directory owner complete access to his or her files. Other users in the directory owner's group have limited access; all other users are allowed no access at all. For instructions to change the parameters from the default, refer to the **Directory Information** chapter under the "File Protection" section.
- QDirectory** Gives information about deleted files.
- RDirectory** (Read Directory) Lists file names together with the date each file was last read. Files are in order of most recent reading.
- Rename** To rename a file:
Ren← OldFileName← NewFileName←(no ≈)
- Set Directory Password**
Passwords are assigned at the time of directory creation. It is always a good idea to change it as soon as possible. To do so, follow this procedure:
Set← Dir← Pas← DirectoryName←
Old password: OldPassword←
New password: NewPassword←
Retype new password: NewPassword←
[*****Unusual passwords are strongly recommended for security reasons, and you should never divulge your password to anyone else for any reason]
- TDirectory** (Time-ordered Directory) Lists the names and write dates of files in the order of the date and time they were last changed.
- Type** To see the contents of a file on your terminal screen:
Type FileName←
Note that this will generally give you garbage with files having the extensions .EXE or .IMP.
- VDirectory** (Verbose Directory) Lists the names of files, protection, size, the date and time they were last changed, and the writer (no headings).
- WDirectory** (Write Directory) Lists the names of the files in order of write date, with the most recently written file first. The writers and creators of the files are given as well.

11.3. SYSTEM-RELATED COMMANDS

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ↵ = Carriage Return

Finger This command will print an alphabetical list of everyone who currently has a job on the system, along with each user's personal name, job number, the number of the terminal she/he is currently logged in on, and the name of the program she/he is using at that moment. Information about a specific individual user can be obtained by including that user's directory name after the Finger command, i.e.

finger DirectoryName↵ .

Information Batch

This command will display a list of all the jobs currently in the batch queue for the user's logged-in system, along with the time that each job is scheduled to run. Since this is sometimes a lengthy queue, a user can qualify this command by typing information batch/user:DirectoryName if she/he wants to see only those jobs submitted by a particular user (DirectoryName).

Information Downtime

This command will display the System's regularly-scheduled "downtime," plus any additional downtimes which may have been scheduled. The NIC system is scheduled to be "down" one night each month (usually the fourth Thursday of the month) for hardware preventive maintenance. When a System is down, it is unavailable for logins.

Information Output

This command will produce a list of all the printing jobs currently in the lineprinter and Imagen (laserprinter) queues for the user's logged-in system. To gain an idea of how soon you may expect your output, type i o↵

Systat

Typing this command will provide the user with a complete list of everyone currently logged in on the system, their job numbers, terminal line numbers, programs in use for each user, and whether each user is logged in directly or is connected from a foreign host. To gain information about an individual user, type sy DirectoryName↵

12. MESSAGE SYSTEMS

MM is the most popular of the message-handling program available on the NIC TOPS-20 system. We recommend that you use MM for sending and reading mail because of its speed and local support.

The program **HERMES** is also available for reading and sending mail, but since it is not as widely used, its level of support is not as great. If a user is interested in learning more about this alternate message program, refer to DOC:HERMES.DOC or type help hermes at the EXEC for more information.

13. MM

MM⁵ is a sophisticated program used to create and manipulate messages. The commands included here are only a small sample of the commands available, but they will allow you to create, send, read, and file your mail on the System.

The figure below shows a typical message. Each part of the message is called a "Field" (To Field, Subject Field, Date Field, etc.). The Fields shown in this particular message (ending with and including the To Field) are what make up the message "Header." Sometimes, you may wish to use the Headers command to see only the header information in a message.

```

Message 1 (429 characters):
Date: Mon 8 Apr 85 17:02:06-PST
From: Steve Dennett <DENNETT@SRI-NIC.ARPA>
Subject: Scribe questions
To: Stacia@SRI-NIC.ARPA
Cc: Roode@SRI-NIC.ARPA

```

Stacia,

Can you see me tomorrow about a few Scribe questions I have?

Thanks,

Steve

Figure 13-1: Sample message

More complete MM documentation is available on the System in DOC: in the files MM.DOC, MM.INTRODUCTION, and MM.PRIMER.

⁵MM was written by Michael McMahon, with contributions from several other individuals. MM is currently maintained by McMahon and Mark Crispin at Stanford University (MRC@SU-SCORE).

13.1. MM - PART ONE

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ↵ = Carriage Return

MM Enter MM from the Exec level by typing MM↵ .

MM> MM> is the MM prompt. Whenever you see MM> on the left side of the screen, you are in MM.

Mail.Txt The file Mail.Txt contains your current messages. It is created by the System when you receive your first message. When you enter MM, this file is automatically loaded into MM for you to read or modify.

Headers All To see your message headers (message number, date, sender, and subject) in the order that the messages arrived, type H A↵ . To see your headers in reverse order (most recent messages first), type H I↵ . To see other commands that can be used with Headers, type Headers ?↵ , and MM will print a list of options.

Read To read a message, type R followed by the message number (for example, R 5↵ to see message 5). To read your new messages, type R↵ . You will now be in Read Mode and the prompt on the left margin will be R> . To see the commands available in Read Mode, type ? at the R> prompt. To continue reading new messages after you finish with the current message, type another ↵ to the R> prompt. To leave Read Mode, type ↵ .

Quit To leave MM and return to Exec, type Q↵ .

↑N To abort the timeout of your Headers or of a message, type ↑N.

MM.Init This is a file that allows you to customize some aspects of MM. Create this file by entering MM and typing Create≈↵ . Exit MM, and use a text editor to edit MM.init. For information on what your options are and how to set each option, read DOC:MM.Info.

? Type ? to see the commands that are available at any point in MM, and a list will be printed on your screen. Once the list is printed, you will be left where you were when you typed the ?.

Help For assistance with any of the commands listed when you type ?, type help Command↵ . For example, type MM> help read↵ for information on the Read command.

General Information About MM

Three of the Exec level commands which also work in MM are:

- ↑S Use ↑S when you wish to hold output of a message on the screen.
- ↑Q When the timeout of a message has stopped because the screen is full or to continue timeout after stopping output with ↑S, type ↑Q to continue.
- ↑T Type ↑T to see the current load average of the System.

For all commands that require a "message number sequence," you may give a single message number, give several numbers separated by commas (for example, read 1,3,7) or, for consecutive messages, the first and last numbers separated by a colon (for example, delete 5:7). Other ways of specifying message number sequences are described in MM.

CAUTION: Never edit a mail file with a text editor. Always use MM commands to delete or edit messages.

13.1.1. Sending a Message

To use the Send Command, type the UNDERLINED words as follows:

MM>S⇐

To: DirectoryName ⇐ To send a message to more than one person, separate each DirectoryName with a comma. Don't use <> around the DirectoryNames.

cc: DirectoryName ⇐ This is for carbon copies. Put your own DirectoryName here if you want a carbon copy of the message.

Subject: Type your subject here⇐

Msg:

Whatever you type here will be the body of your message. Be sure to type a ⇐ near the end of each line to make your message more readable.

To end your message, type ↑Z.

13.1.2. Send Mode Commands

Once you type ↑Z to end your message, the prompt on the left side of the screen will be S> because you will be in Send Mode. Now you may use any of the following commands (type ? to see additional commands that can be used here):

Send To send your message, type ⇐ You may edit your MM.init file so that you must type Send⇐ to send your message.

Quit To abort your message, type Q⇐ .

Display To see your entire message before sending it, type Display⇐ .

Edit To edit your message in EMACS, type Edit⇐ and you will enter EMACS with the

body of your message in the main buffer. You may edit your message and then return to MM by typing ↑X↑Z.

From Adding a From: Name field indicates that you are sending a message for someone else. The name does not have to be a DirectoryName. Your name will still appear in the message header as "Sender." Do not type the colon when you are adding this field. For example:

S> from ≈ (NAME) Stacia Snapp

Storing and Retrieving Messages before Sending

If you have created a message but you do not want to send it immediately, you may store it as a file in your Directory and then retrieve and send it at a later time. To store your message before sending it, type (at the S> prompt)

S> Save-Draft ≈ FileName ←

S> q ←

To retrieve your stored message, type

MM> Restore-Draft ≈ FileName ≈ ←

and you will be in the text field of your message. You may add to your message or type ↑Z to end it, and then edit or send it.

Commands for Corrections

Del Key To delete one character at a time moving from right to left (may need the Shift Key).

↑W To delete one word at a time moving from right to left.

↑U To delete the entire line from the cursor to the left side of the screen.

13.2. MM - PART TWO

Key:
 ↑ = Control Key
 ≈ = Escape Key
 # = Message Number
 ⇐ = Carriage Return

- Delete** To delete the current message, type Del #⇐ . To delete a sequence of messages, type Del and the numbers of the first and last messages in the sequence separated by a colon (:) (e.g., Del 10:15⇐ deletes messages 10 through 15). To delete messages not in sequence, separate each message number or group of messages with a comma. Deleted messages can still be retrieved with the Undelete command as long as they have not been Expunged (and you have not Exited from MM).
- Undelete** To undelete a deleted message, type Und #⇐. You may also undelete more than one message at a time by specifying a message sequence (as described in Delete above).
- Expunge** To permanently remove all deleted messages, type Expunge⇐ . Your remaining messages will be renumbered and you will be left in MM.
- List** To get a lineprinter copy of a message, type List #⇐ Type help hardcopy⇐ for a description of other methods for getting a printed or "hard copy" of your message on the lineprinter or Imagen.
- Reply** To reply to a message you have received, the command is Reply #⇐ (where # is the message number you wish to respond to). MM will prompt you, "Send reply for message # to:" and you may type s⇐ to reply to the sender only, or type a⇐ to send a copy to everyone listed in the "To" and "Cc" fields of the original message. The "To," "From," and "Subject" fields will be completed by MM, and you will be prompted for "Msg." Type your reply and end the message with ↑Z. Then type S>S⇐ to send it.
- Forward** To send a copy of a message you have received to someone else, type Forward #⇐ . You will be prompted for "To" and "Msg." If you want to add a message, type it here and end with ↑Z. To see the entire message before sending it, type (after ↑Z) S>Display⇐ . Then type S>S⇐ to send it.
- Remail** This command allows you to send a copy of a message you have received to someone else without adding anything to the message (as in Forward). Type Remail #⇐ . You will be prompted for "To" and your message will be sent immediately after you type ↑Z.
- Move** This command is useful for collecting related messages into one file. To move messages from your Mail.Txt into another message file, type Move≈FileName≈ #⇐ If this is a new file, type Move≈FileName #⇐ . Use the Get command to see other message files.
- Get** To read a mail file other than Mail.Txt into MM, type Get FileName≈⇐ To return to your Mail.Txt, type Get⇐ To read the System's general BBOARD, you need only type BBOARD⇐ To read "special interest" bboard files, type BBOARD FileName⇐. See the description of BBOARD Special Interest Files in the chapter titled "Mail-Related Programs."

14. MAIL-RELATED PROGRAMS

The programs listed below are not message systems, but are related to the reading, addressing, and/or delivery of electronic mail. Each program is followed by the filenames which contain online information.

- BBOARD** BBOARD is a program for reading bulletins that have been placed in the <BBOARD>MAIL.TXT by users. It keeps track of the messages a user has viewed and the ones that are still unseen and allows the user to selectively examine the various messages that have been posted since he or she last examined the file. A more detailed description of the program can be found with help bboard and in the file DOC:BBOARD.DOC.
- <BBOARD> Special Interest Files**
In addition to the MAIL.TXT file in the <BBOARD> directory (described above), there are "special-interest" files which contain communications from users across the ARPANET and MILNET about particular topics such as the IBM Personal Computers, the APPLE computers, and the UNIX operating system. These files are named in the format *filename.txt*. You can see a list of the current special-interest files by referring to <BBOARD>-NIC-INTEREST-GROUPS.TXT.
- You may use MM to view the mailing list files (using the "get" command), or you may type at the EXEC: BBOARD FileName. This runs the program BBOARD.EXE, which will display the messages you have not yet seen in the file. To read messages dated earlier than those found online, refer to <BBOARD>INTEREST-GROUPS.TXT and send a message to the appropriate contact.
- MAILSTAT** MAILSTAT is a utility for showing queued mail. It queries a system job which then checks for your queued mail in the directory you are connected to and any queued mail you might have in the system mail directory, MAILQ:. Since the program queries a system job, ^C and ^O will not work to stop its output.
- MMAILBOX** MMAILBOX is a program which accesses the mail-forwarding database and provides the net-address for a specified directory. To invoke this program, type mmailbox at the EXEC level. You will then be prompted for an **Address:**. Respond by typing in the name of a user or directory name, followed by a carriage return. The output will be the directory name followed by the host on which that directory has its primary mailbox.
- REMIND** This program can be used to send messages to yourself and/or other users at a specified time in the future. Type help remind at the EXEC, or read the file DOC:REMIND.DOC for instructions on using this valuable program.

15. TEXT EDITING PROGRAMS

The most popular text editing programs (and the ones that we recommend) are EMACS, a screen-oriented editor, and EDIT, a line-oriented editor.

The default editor on the NIC system is EMACS. You can specify which editor you want to use by default with the following command:

```
@def ≅ editor: ≅ EditorName←
```

where EditorName is SYS:EMACS or SYS:EDIT.

16. EMACS

Emacs⁶ is a very powerful text editor. It is a **screen editor**, so the lines are not numbered. All commands start with either the Escape Key or the Control Key. When you use the Escape Key, type it once, release it, then type the next character. When you use the Control Key, hold it down while you type the desired character. (Using the Control Key to obtain a "control-character" is similar to using the Shift Key to get a capital letter.) Anything you type without the Escape Key or the Control Key will be entered as text. The cursor shows where text will be entered if you type, and the cursor can be moved anywhere on the screen.

In the instructions that follow, there will be several references to "regions." A region is simply a portion of a file which has been marked off by the user for purposes of deleting, moving to a different location in the file, etc.

Near the bottom of the screen is a line called the "Mode Line" containing information that is continuously updated. It may show the time and load average, the name of the file you are currently working on, where you are in the file at that moment, and whether the file has been changed since you last updated and saved it.

The area of the screen below the mode line often shows where you are in a command. For example, whenever you type the Escape Key, the characters "M-" appear on that line.⁷ When you see "M-" you know that Emacs is going to consider the next character you type to be a command. It is important to pay attention to this part of the screen.

⁶Emacs was written by Richard M. Stallman at the MIT Artificial Intelligence Lab, and is distributed on a basis of communal sharing. AI Lab memo 519a is available for people who want more information on how EMACS works.

⁷Emacs was written for terminals having a Meta Key. It was altered to work on terminals without a Meta Key so that the Escape Key works instead. Whenever you see "M-" on your screen or in Emacs documentation, it may indicate the Escape Key if your terminal does not have a Meta Key.

16.1. EMACS - BASIC COMMANDS

Key:
 ↑ = Control Key
 ≅ = Escape Key
 ⇐ = Carriage Return

Help To read the help options, type the control key and the underscore key simultaneously and follow the instructions to get information about command characters, functions, Emacs news, etc.

Enter Emacs Emacs⇐

Create New File
 To create a new file, enter Emacs and start typing.

Name New File
 ↑X↑W NewFileName⇐ This command also writes out (saves) the file.

Update/Save File
 ↑X↑S This command creates a new generation of the file you are working on.

Load an Existing File
 ↑X↑F FileName⇐

Leave Emacs ↑X↑Z Returns you to EXEC level. Be sure that you have saved the current generation of the file (if you want it) before exiting.

Abort Command
 Type ↑G to abort a command that has not yet been completed. Sometimes, you must type several ↑G's to abort.

Undo Last Command
 Type ≅X undo⇐ to actually undo a command immediately after it is completed. If you mistakenly give a command that makes a great change in the text buffer, by using this command you can often undo the change without having to know precisely how it came about.

16.2. EMACS - BASIC CURSOR COMMANDS

Key:
 ↑ = Control Key
 ≈ = Escape Key

DELETING TEXT

<u>↑D</u>	Delete next character
<u>≈D</u>	Delete next word
<u>Del</u>	Delete previous character (may need the Shift Key)
<u>≈Del</u>	Delete previous word (may need the Shift Key)

MOVING THE CURSOR

<u>↑F</u>	Move forward one character
<u>↑B</u>	Move back one character
<u>↑N</u>	Move to next line
<u>↑P</u>	Move to previous line
<u>↑A</u>	Move to beginning of line
<u>↑E</u>	Move to end of line
<u>≈A</u>	Move to beginning of sentence
<u>≈E</u>	Move to end of sentence
<u>≈<</u>	Move to beginning of file
<u>≈></u>	Move to end of file
<u>≈F</u>	Move forward one word
<u>≈B</u>	Move back one word
<u>↑V</u>	Display next screen of text
<u>≈V</u>	Display previous screen of text

MISCELLANEOUS COMMANDS

<u>↑O</u>	Inserts blank line at the current cursor position
<u>↑L</u>	Clears screen and redisplay it
<u>↑T</u>	Transposes characters before and at the cursor
<u>≈T</u>	Transposes words before and after the cursor
<u>≈C</u>	Changes first letter of word to uppercase (Don't type ↑C by mistake!)
<u>≈U</u>	Changes word to uppercase
<u>↑X↑U</u>	Changes region to uppercase
<u>≈L</u>	Changes word to lowercase
<u>↑X↓L</u>	Changes region to lowercase

16.3. EMACS -- ADDITIONAL FEATURES

Key:
 ↑ = Control Key
 ≅ = Escape Key
 ⇐ = Carriage Return

Entering Control Characters as Text

If you need to enter a control character into your file as text (rather than typing it as a command), type ↑Q and then type the control character you are entering. The ↑Q will not appear on your screen. For example, type ↑Q↑L to enter a pagemark in your file. **Note:** Using ↑Q will not work for inserting a ↑C.

Exchanging Text (Query Replace)

This command finds each successive occurrence of a string of text and allows you to decide whether or not to exchange it for another string of text. Type ≅% existing text (give current string to be changed) ≅ new text (give desired string) LL to start the exchange. As each occurrence of existing text is found, you may have it replaced with new text by typing one space bar or skip it by typing Del Key (you may need the Shift Key). You may confirm all exchanges by typing !. To abort the command, type ≅.

Joining Files

To combine files, read one file into a buffer with ↑X↑V, move the cursor to the point where you want the second file to be, and then read the second file into the same buffer with ≅X Insert File FileName⇐. The second file will be inserted immediately below the cursor.

Killing Lines and Regions

To delete ("kill") one line, go to the beginning of the line and type ↑K. To kill a region (anything longer than a line), first mark the beginning of the region to be killed by moving the cursor to the first character of the region and typing ↑@ (the @ won't appear on the screen). Then move the cursor one character past the end of the region to be killed and type ↑W. To retrieve what has been killed, type ↑Y.

Minibuffer

If you should happen to type ≅ twice accidentally, or if you hold the Escape Key down too long, you will enter the "Minibuffer." To get out of the Minibuffer, type ≅ twice again, and you will be back where you were in your file. The Minibuffer has some useful applications for certain complicated commands; however, extensive use of the Minibuffer is recommended for experienced Emacs users only.

Multiple Buffers

A buffer is a working space in Emacs. While you are editing a file, it is in a text buffer. Each buffer holds one file and has a separate name (usually the first part of the FileName). Emacs allows you to have multiple buffers and to switch back and forth between them. To read a file into a buffer, type ↑X↑F FileName⇐. To see the names of the buffers and the names of the files in them, type ↑X↑B (notice the buffer names). Type the space bar once to get back to your file. To switch from one buffer to another, type ↑XB BufferName⇐.

Key:
↑ = Control Key
⌘ = Escape Key
↵ = Carriage Return

Repeating Commands

Most commands can be told to repeat a specified number of times. To repeat a command, type ⌘, then type the number of times you wish to repeat the command, then type the command. For example, to move forward 1 word in a file type ⌘F and to move forward 10 words type ⌘10 ⌘F.

Searching For Text (Incremental Search)

To search forward in your file, type ↑S and then type the text you're searching for. End or abort the search by typing ⌘. To find successive occurrences, repeat ↑S. To search backward in your file, use ↑R.

16.4. EMACS -- LIST OF ALPHABETIC COMMANDS

Key:
 ↑ = Control Key
 ≈ = Escape Key

↑A	Moves to beginning of line
≈A	Moves to beginning of sentence
↑B	Moves cursor back one character
≈B	Moves cursor back one word
↑C	Exits EMACS without saving file!
≈C	Capitalizes next word
↑D	Deletes next character
≈D	Deletes next word
↑E	Moves to end of line
≈E	Moves to end of sentence
↑F	Moves cursor forward one character
≈F	Moves cursor forward one word
↑G	Aborts an unfinished command
↑K	Kills current line
↑L	Clears screen and reprints it
≈L	Changes word to lowercase
↑N	Moves cursor to next line
↑O	Inserts blank line above current line if cursor is at the beginning, or breaks a line into two lines at the cursor
↑P	Moves cursor to previous line
↑Q	Allows you to enter control characters as text
↑R	Reverse incremental search
↑S	Incremental search
↑T	Transposes characters before and at the cursor
≈T	Transposes the words before and at the cursor

Key:
↑ = Control Key
≈ = Escape Key

↑U	Causes the next command to be executed 4 times
≈U	Changes word to uppercase
↑V	Displays next screen of text
≈V	Shows previous screen of text
↑W	Kills region between cursor/mark
≈W	Copies region into kill buffer
↑Y	Unkills last killed text

18.5. EMACS -- LIST OF OTHER USEFUL COMMANDS

	<u>key:</u>
	↑ = Control Key
	≈ = Escape Key
↑ _	Offers help
≈ <	Goes to beginning of file
≈ >	Goes to end of file
↑ 0	Marks beginning of region to be killed (in combination with ↑ W)
↑ X ↑ F	Finds file (user supplies name) and places it in a new buffer
↑ X ↑ L	Changes region to lowercase
↑ X ↑ S	Saves file (write new version of file using current name)
↑ X ↑ U	Changes region to uppercase
↑ X ↑ V	Visits file (user supplies name) and places it in current buffer
↑ X ↑ W	Writes file (user supplies name)
↑ X ↑ X	Exchange cursor and mark (used in combination with ↑ 0)
↑ X ↑ Z	Exits Emacs without saving file

17. FTP -- FILE TRANSFER PROTOCOL

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

The FTP program allows users to transfer files over the ARPANET or MILNET. In preparation for the file transfer, it is necessary first to connect to the host from (or to) which files are to be transferred. Once the connection is made, you should login, supplying *DirectoryName* and password (see the following examples). Note that you are actually logging into the host that you connect to, so you must have a directory on that host that you can log into (either your own or a guest account).

Once you have logged in, you're ready to begin the file transfer process. A list of available commands may be obtained by typing ? at the FTP command prompt (FTP>). This document shows two examples of the simplest file transfers.

The two commands used to transfer files are **GET** and **SEND**. The **GET** command is used to retrieve a *RemoteFile* and copy it to a *LocalFile*. The **SEND** command is used to send a copy of a *LocalFile* to a *RemoteFile*. A *LocalFile* is a file residing on the host on which you are running the FTP program. A *RemoteFile* is a file residing on the host that you've connected to through FTP. The MULTIPLE GET/SEND commands permit the user to transfer multiple files at one time, a very useful feature of FTP. For further information, type ? at the FTP command prompt (FTP>).

After transferring the files, close the current connection by typing the **BYE** command. The **BYE** command breaks the connection "cleanly." Then type the **QUIT** command to exit from the program and return to the EXEC level of your logged-in system. You may also simply use the **EXIT** command to do both of these (break the connection and return you to the Exec).

The examples on the following page illustrate simple **SEND** and **GET** commands. The user should type what is UNDERLINED. For more information, type help ftp⇐ or see the file DOC:FTP.DOC.

Key:
 ↑ = Control Key
 ⌂ = Escape Key
 ↵ = Carriage Return

```
@ftp␣
FTP>␣␣␣
< SRI-KL.ARPA FTP Server Process 5Z(14)-7 at Wed 10-Apr-85 14:51-PST
Setting default transfer type to paged.
FTP>log␣(user) DirectoryName␣
Password: Password␣
< User DirectoryName logged in at Wed 10-Apr-85 14:52-PST, job 23.
FTP>send␣(local file) LocalFile␣(to remote file) RemoteFile␣
PS:<DirectoryName>LocalFile ==> PS:<DirectoryName>RemoteFile;P775202 !! [OK]
FTP>bye␣(break connections with remote host) ␣
FTP>quit␣(to EXEC) ␣
⓪
```

Figure 17-1: Example of FTP SEND Command Sequence

```
@ftp␣
FTP>␣␣␣
< SRI-KL.ARPA FTP Server Process 5Z(14)-7 at Wed 10-Apr-85 14:53-PST
Setting default transfer type to paged.
FTP>log␣(user) DirectoryName␣
Password: Password␣
< User DirectoryName logged in at Wed 10-Apr-85 14:52-PST, job 23.
FTP>get␣(remote file) RemoteFile␣(to local file) LocalFile␣
PS:<DirectoryName>RemoteFile ==> PS:<DirectoryName>LocalFile;P775202 !! [OK]
FTP>exit␣(and close connection) ␣
⓪
```

Figure 17-2: Example of FTP GET Command Sequence

18. THE TN PROGRAM

Key:
 ↑ = Control Key
 ⌘ = Escape Key
 ↵ = Carriage Return

Connecting to Another Computer

The TN Program allows you to login on one computer, connect to another computer, and then login on that one. In order to do this, you must have directories on both machines that you may use to login. To use TN to connect to SRI-KL from SRI-NIC, for example, you would type (underlined words are what you type)

```
@tn sri-kl↵
Trying... Open

SRI-KL, TOPS-20 Monitor 5.3(5715)
There are 54+3 jobs and the load av. is 1.20, TTY247.
@log DirectoryName Password↵
Job 56 on TTY337 15-Apr-85 11:03AM
Previous LOGIN: 10-Apr-85 1:49PM
@
```

Now you are logged in on SRI-KL. To return to SRI-NIC, simply logout from SRI-KL and the connection to SRI-KL should be broken by the TN program.

```
@logout↵
System shutdown scheduled for Wed 26-May-82 10:00PM,
Up again at Thu 27-May-82 4:00AM
Killed Job 56, User DirectoryName, Account, TTY 122,
at 26-May-84 11:04:32, Used 0:00:01 in 0:00:32

Connection closed by foreign host
@
```

Now you are back on SRI-NIC. If for any reason the connection is not broken, type ↑↑C (hold down the control key, type the up arrow key, and then type C). This will also break the connection.

Checking Where You Are

If you are uncertain about what machine you are on, type at the EXEC

```
@i ver↵
```

and the System will print something like

```
SRI-NIC, TOPS-20 Monitor 5.3(5743)-1
TOPS-20 Command processor 5.1(1367)-1
```


19. THE ARCHIVE/MIGRATION SYSTEM

A Directory is like a file cabinet in that it has only a limited amount of storage space for files. Any effective filing system will observe certain procedures of regular maintenance, particularly the following four:

1. Adding files
2. Retaining current and pertinent files
3. Discarding unnecessary and outdated files
4. Moving inactive files (that may be needed later) to remote storage

The information that follows will explain how you and the System work together in maintaining your Directory (your "computerized file cabinet") through the Archive/Migration System.

19.1. ARCHIVING - OVERVIEW

Archiving is a software system which allows users to move files from their disk areas to magnetic tape. If you have a file or files in your directory that you do not need now but want to keep, you can mark them for permanent off-line storage by using this system.

The operator periodically runs a program that copies the files marked for archival from disk to magnetic tape. When the copy has been made, the disk file is usually deleted. You can prevent the disk copy from being deleted by using the RETAIN option of the archive command. However, in most cases it doesn't make sense to archive a file unless you want to remove it from the disk to save space.

Currently all archiving is done to pairs of tapes. Your file will not be removed until it has been successfully written to two tapes. These tapes are kept indefinitely.

To request that your file be moved to tape, use the Archive command. Under normal circumstances your file will disappear from your directory as soon as you have requested it to be archived. Actually, the file is still there until the operator gets around to writing it to tape. However it is given "invisible" status so that it no longer clutters up your directory. Once the system has written the second copy to tape, it will normally remove ("delete") the contents of the file from disk. Directory information (such as file name, creation date, and the names of the tapes on which the file has been archived) will be retained. The result of this process is a file which is "offline," "archived," and "invisible".

You can mark your files for archival by typing:

```
@Archive≡ FileName ←
      FileName [requested]
```

19.2. MIGRATION - OVERVIEW

The migration software allows the system staff to move files which have not been touched recently to magnetic tape. This frees up disk space, an expensive system resource. To migrate files, the operator runs a program which goes through directories on the computer, marking for migration files which have not been accessed within a certain period. The operator then runs another program which moves those marked files onto magnetic tape.

As with archiving, the file must be written successfully onto two tapes before being removed from disk. After the files have been migrated, the System will use electronic mail to notify affected users that migration has taken place and inform them which of their files were moved.

Because migration is a **system-initiated** rather than a **user-initiated** process, the designers of Tops-20 felt that users should be left with an easily seen reminder of the affected files. Thus, unlike archived files, migrated files are left in a "visible" state. In a directory listing, the file or files migrated will show up with the comment ";OFFLINE" next to the filename.

19.3. MIGRATION - POLICY

On the NIC 2080 forced migrations are done on the first of every month. The reap period used on the NIC is 122 days. This means that files which have not been referenced for 122 days will be moved offline at the beginning of each month. This includes both previously archived files and regular files. A week before the migration is done, electronic mail messages are sent to users with files which would be affected by the migration.

19.4. COMMANDS - DESCRIPTIONS AND EXPLANATIONS

19.4.1. GETTING ARCHIVE-STATUS INFORMATION

To check whether a file is marked for archival, give the Information Archive-Status command, followed by the filename (shown in the first of the following examples).

If the Information Archive-Status command is given without specifying a filename (as in the second of the following examples) the System will print a list of:

1. Files which have been archived
2. Files for which archiving has been requested
3. Files for which migration is resisted (not useful on the NIC)
4. Files for which migration is prohibited (this status can only be set by privileged users)

- Examples -

1. The filename is specified in this example.

```
@Information ≈ archive-status ≈ FileName ≈ ←
  FileName archive requested
```

2. In this example, the filename is not specified; therefore archive/migration-status information is provided for all files in the directory.

```
@Information ≈ archive-status ≈ ←
  FileName Archive Requested
  $USER-PROFILE$.DW5.1 Migration Resisted
  MAIL.TXT.1 Migration Prohibited
```

19.4.2. CHECKING ON FILES TO BE MIGRATED

You can check to see which of your files would be migrated by using the Reaper program. This will give you a list of the files that would be migrated were the migration being done on the day that you run Reaper.

For example:

```
@reaper←
Output to: tty:←
Check files: *.*←
```

REAPER run started at 12-Apr-85 15:00:54

Specified file path: PS:<VIVIAN>*.*

The following would happen if the OPERATOR ran REAPER now:

```
PS:<VIVIAN>
CHECK.PAS.16          5    1-Dec-84
RESUME.CURRENT.2      2    4-Dec-84
```

Total of 7 pages in 2 files.

```
2 files marked for migration, 7 pages
0 archive files deleted from disk, 0 pages
0 temporary files deleted, 0 pages
0 expired files purged, 0 pages
```

If you are a privileged user and run Reaper when **enabled**, you can potentially do damage to the file system since Reaper puts enabled, privileged users in command mode rather than doing a scan-only.

Please note that you will not necessarily see all the files which would be migrated unless you are running Reaper after 4:00pm of the day of the migration. You can get a more accurate estimate of the files which would be affected by a migration if you calculate the date which 122 days before the first of the next month, and then do a directory command with the subcommands before, online, and reference. For example, these are the commands you would use if you wanted to see the files which would be affected by a migration done on May 1, 1985

```
@directory, ←
@@before 30-dec-84 ←
@@times read ←
@@chronological ←
@@online ←
PS: <VIVIAN>
```

Rd

```
CHECK.PAS.16      1-Dec-84 12:49:42
RESUME.CURRENT.2  4-Dec-84 01:31:40
FORMLE.EXE.1      7-Dec-84 17:10:18
LABELS.EXE.1      7-Dec-84 17:10:36
FORM.CREATES.4    11-Dec-84 14:25:28
POWEROUTAGE.LIST.3 12-Dec-84 14:05:05
FORM.SAI.2        22-Dec-84 13:59:19
INC-FORM..2       1-Jan-85 14:16:36
HOG.CTL.1         2-Jan-85 19:37:30
ARMAIL.REL.1      24-Feb-85 16:58:31
FORM.TOPS20-CREATES.1 26-Feb-85 12:25:07
DMPLAB.EXE.4      26-Feb-85 16:42:17
VISITOR..1        27-Feb-85 13:13:47
TTY-CONFIG.TXT.55  3-Mar-85 19:09:53
IS.EXE.1          13-Mar-85 17:55:42
EMACS.MINE.1      20-Mar-85 15:12:48
ULTCMD.FAI.10     28-Mar-85 23:16:33
SPEECH.1.1        29-Mar-85 16:19:36
MSGFIX.EXE.1      1-Apr-85 15:29:14
MSG.TXT.1         12-Apr-85 15:40:40
F4.DIRECTORY.1    13-Apr-85 20:19:14
```

Total of 9 files

The files with a read date before 30-Dec-84 are the ones that would be migrated on 1-May-85, so the files from CHECK.PAS to FORM.SAI would be migrated in this example.

19.4.3. CANCELING AN ARCHIVE REQUEST

You can cancel an archive request before the archive is run by typing:

```
@cancel ≈ archive ≈ FileName ←
      FileName [OK]
```

19.4.4. RETRIEVING AN ARCHIVED OR MIGRATED FILE

To bring back a file which has been archived, use the Retrieve command. This command notifies the System that you want a file moved from magtape back into your directory on disk.

```
@retrieve≈ FileName≈  
           FileName [OK]
```

Retrieval requests should be processed within a half hour of when they are made.

19.4.5. CHECKING THE STATUS OF YOUR RETRIEVAL REQUEST

To see your retrieval request, give the Information (about) Retrieval-Requests command. The System will respond with a list of all retrieval requests currently in the retrieval queue.

```
@info≈ retrieval≈  
Retrieval queue:  
Name      Req#      Tape1  Tape2      User  
-----  
MYTEST    24         5803   5807      DWILLIAMS
```

Once an archived file is restored to disk, that generation of the file can only be read. Therefore the file can only be modified if the changed version is written to a new generation of the file or a different file name altogether. This is a built-in safeguard to prevent users from accidentally destroying the tape storage information in the file descriptor. Migrated files do not have this safeguard, so they may be edited (after being retrieved) as a normal file.

CANCELING A RETRIEVAL REQUEST

To cancel any retrieval request before the contents of the archived file are restored, use the Cancel Retrieve command.

```
@cancel≈ retrieve≈ FileName≈  
[1 Job canceled]
```

DISCARDING ARCHIVE INFORMATION

If you accidentally mark a file for archive or decide that you would like to write on the same version of a file that has been archived but is now online, you can use the Discard command to restore the file to a "normal" status. The limitations on this command are that the file modified with it must be online and visible. Following is an example:

```
@info≈ archive-status≈ login≈  
LOGIN.CMD.15 Archive requested  
@discard≈ login.cmd.15  
LOGIN.CMD.15 [OK]  
@info≈ archive≈ login≈
```

As you can see the discard command also has the side effect of cancelling an archive request for a file that has not been moved to tape yet.

DELETING AN ARCHIVED FILE

1. In this example, the archived file (whether online or offline), is being deleted, as well as the System's record of where that file can be found on tape. When this kind of deletion is done, the System will send the user a message giving the filename and tape information that is usually given when the user is requesting a file for retrieval. This is your final record of the file's location on tape. This method applies for files that have been archived under version 4. The file must be visible, so type Set File Visible *FileName* if the file is still invisible.

```
@delete~ FileName,←
@@archive←
@@←
```

2. Here, the contents of the retrieved archived file are deleted, but the filename remains in the directory and the tape information is retained.

```
@delete~ filename,←
@@contents-only←
@@←
```

DELETING A MIGRATED FILE

Deleting a migrated file is done by simply using the Delete command specifying the filename as you would with any file you wish to remove. The danger of deleting a migrated file (whether online or offline) is that when the file is deleted, the record of the tape copy is gone as well.

SETTING OFFLINE FILES INVISIBLE

If you prefer not to see migrated files each time you issue the directory command, they may be set invisible. There is a PCL command which will allow you to set all OFFLINE files invisible with a single command. The command file is PS:<PCL>OFFREN.PCL, and it creates a command called OFFLINE-INVISIBLE. On the following page is an example on how to declare and use this command.

```

@declare~ pcl-routines ps:<pcl>offren~←
[Command Offline-Invisible defined]
@directory,←
@@offline~←
@@←

```

```

PS:<VIVIAN>
INTERVIEW.QUESTIONS.3;Offline
SELF-EVALUATION.1984.1;Offline
X-MAS.1984-SCHEDULE.1;Offline

```

Total of 3 files

```

@offline-invisible~ *.*←
INTERVIEW.QUESTIONS.3 [OK]
SELF-EVALUATION.1984.1 [OK]
X-MAS.1984-SCHEDULE.1 [OK]
3 files set invisible

```

VOCABULARY

Archive	Permanent, offline (magtape) storage of a file. Files are archived only when marked for archival by the user. Marking a file for archive will prevent that version of the file from being modified. If you want to change it, you have to make a new copy or Discard the tape information from the file descriptor. This status is permanent, even if you retrieve the file. Files are moved at least every two weeks. To check the last time that files were moved, type <u>help status</u> ←
Invisible	Files with the invisible attribute will not show up in the usual Directory listing unless a special subcommand "invisible" is given. When files are marked for archive, this attribute is automatically set. The <u>@Set File Visible</u> command will make the file visible again by turning that bit off.
Magtape	Short for magnetic tapes. Archive magtapes are located onsite at the NIC in the computer room and the vault.
Migrate	Offline (magtape) storage of a file. Migration takes place independent of user request.
Online	A file which is on disk and readily accessible.
Offline	A file which has been removed from the disk and placed in storage on magtape. The directory information is still present on the System. However, the contents are only on tape.
Prohibit-Migration	This attribute tells the System not to migrate the file. Prohibit-Migration can only be set by privileged users. It is automatically set on all MAIL.TXT files. You can request to have this attribute set by sending a message to ACTION with the FileName and reason.
Retrieve	This command is used to bring back a file from offline storage (magtape) into the user's directory (online storage). It can be used to bring back files that have been either archived or migrated.
Resist-migration	This attribute tells the System not to take the specified file offline when doing over-quota pruning. We do not do over-quota pruning on the NIC so it is not a useful attribute to set.
Visible	Files that show up with the regular Directory command are visible.

20. SCRIBE -- BASIC COMMANDS

Scribe⁸ is a text formatting program. It is used for preparing text to be a report, manual, article, letter, or any other type of document. Scribe processes text files that have formatting directions (commands) in them, and produces new files that can be printed on a variety of output devices. Some examples of the types of formatting commands are the following: italicize (make the following text appear in italics), center (center the following text), underline (underline the following text). Scribe can distinguish commands from text because every Scribe command starts with an at-sign (@). The examples above would look like this: @i, @center, @u. Another Scribe command you should give indicates the output device (Imagen (Imprint10), lineprinter, or terminal screen) on which the file created by Scribe is to be printed. Once you use a text editor to create a file that contains text and Scribe commands, Scribe can process it and produce a second file, formatted according to the commands in the original file that you created.

First, use a text editor to create a file that has text along with Scribe commands. The Scribe commands are typed as if they were ordinary text. Then save the file, leave the text editor, and return to the EXEC level. Enter Scribe, and give it the name of the file that you created. Scribe will read that file, interpret the commands, and create a second file by following the directions (commands) that you put in the first file. Now you may send the file that Scribe created to the appropriate output device.

Scribe is a large, complex program with many features that you may want to use. This chapter provides a general overview of Scribe. Much more detailed information can be found in the Scribe Document Production System User Manual by Unilogic.

⁸"Scribe" is a registered trademark of UNILOGIC, Ltd., Pittsburgh, PA.

20.1. FILENAME TYPES

The FileName Type is the part of the name that follows the "." (period) in the name. With Scribe files, the Type is significant. **The file you create with Scribe commands in it should always have the Type .MSS** (for example, *testfile.MSS*). Scribe processes the .MSS file you create and produces a second file with the same name but a different Type. The Type depends on the printing device you specified in the .MSS file. The list below shows the file Type that Scribe associates with each printing device.

If Scribe detects any illogical commands while it's processing your .MSS file, it will also produce an .ERR file, which contains a list of errors in the .MSS file. When this happens, you must fix the problems in the .MSS file yourself because **Scribe never alters the .MSS file**. After you have fixed the .MSS file, run it through Scribe again.

.MSS

Source file. When you create a file with Scribe commands in it, give it this Type. **This is the only file that you name or edit**. When you run this source file through Scribe, it will automatically be renamed with one or more of the filetypes below.

.LPT

Lineprinter file. When you use @Device[Lpt], Scribe will produce a file with the Type .LPT. The .LPT file must be printed on the Lineprinter.

.IMP

Imagen file. When you use @Device[Imprint10], Scribe will produce a file with the Type .IMP. The .IMP file must be printed on the Penguin.

.DOC

Terminal file. When you use @Device[File] or @Device[PagedFile], Scribe will produce a file with the Type .DOC. You can print this file on your terminal screen using the **Type** command at the EXEC level.

.ERR

Error file. If Scribe finds any errors while it is processing your .MSS file, it prints them on your screen while it works and also produces an .ERR file that you can read with a text editor.

.OTL

Outline file. This file contains an outline of the corresponding .MSS file's chapters, sections, and subsections. Scribe creates an .OTL file only for certain kinds of documents (for example, articles and manuals).

20.2. GLOBAL COMMANDS

Global commands affect the entire file. If these commands are used, they must appear at the beginning of your .MSS file. The three most often used Global commands are "Make," "Device," and "Style."

The @Make Command

If you use the @Make command, it should be the first line of your .MSS file. This command indicates the kind of document you are creating and should be followed by one of the options shown enclosed in delimiters:

@Make[Text]

If the @Make command is not given, it will default to @Make[Text]. This is the simplest document type and makes justified (even right margin) paragraphs.

@Make[Article]

An Article has three levels of numbered headings: Section, Subsection, and Paragraph, as well as Appendix and AppendixSection.

@Make[Report]

A Report has a title page, numbered chapters, sections, subsections, and a table of contents.

@Make[Manual]

A Manual is like a Report, but can also have an index.

@Make[Slides]

Slides uses a large font and appropriate line spacing for overhead transparencies.

@Make[Letterhead]

Letterhead produces a business letter format with the SRI Letterhead.

@Make[MilStd847B]

Creates a document which conforms to the Military Standard format.

The @Device Command

The second line of a Scribe .MSS file should be the @Device command. (If there is no @Make command, the @Device command should be the first line of the file.) This command tells Scribe to create a file that can be printed on a specified device. It should be followed by one of the options shown, enclosed in delimiters:

@Device[File] or @Device[PagedFile]

This tells Scribe to create a .DOC file. You may print a .DOC file on your terminal screen with the Type command at the EXEC level.

@Device[Lpt]

This tells Scribe to create an .LPT file. The .LPT file should be printed on the Lineprinter with the Print command at the EXEC level.

@Device[Imprint10]

This tells Scribe to create a .IMP file. This file should be printed on the Imagen laser printer with the Laser command at EXEC level. If the @Device command is not given, Scribe defaults to @Device[Imprint10].

The @Style Command

Style commands allow you to change the appearance of the entire document. If they are used, Style commands should be grouped together immediately following the @Device command. Some useful Style commands are

- @Style[Spacing 2]** Makes double-spaced text.
- @Style[Justification off]** Makes filled text with unjustified ("ragged") right margin.
- @Style[Indent 0]** Causes the first line of each paragraph to be flush left.
- @Style[Indent 5]** Causes the first line of each paragraph to be indented five spaces.

20.3. USEFUL SCRIBE COMMANDS

Most commands can be in either of two forms:

1. The command followed by "delimiters" surrounding the text to be affected, for example, @center[Scribe Commands], or
2. The command delimited by "begin" and "end," for example,

```
@begin[center]
Scribe Commands
@end[center]
```

It's better to use @begin[Scribe-Command]...@end[Scribe-Command] for multi-line items.

The result of the two examples is the same:

Scribe Commands

Delimiters must match each other within a command. The delimiters you may use are: (...), [...], {...}, <...>, '...', and "...". Choose your delimiters carefully so that the text or figures inside the delimiters do not include the particular delimiters used.

A blank line

A blank line indicates the end of a paragraph.

@center

This command centers text.

@i This command italicizes text.

@u This command underlines text. (See the Scribe manual for a complete description of underlining options.)

@verbatim

Verbatim produces text that is not filled or justified but appears exactly as you typed it. The text in Verbatim is printed in a fixed-width font (similar to a typewriter). Other Scribe commands within Verbatim will still be recognized.

@format

This command is similar to @Verbatim, but the text within @Format stays in the same font as the body of the text.

@quotation

This command produces text that is single spaced, filled and justified, and with wider right and left margins (the margin is the white area on each side of the text).

@verse

Verse produces results similar to @Quotation but does not fill and justify.

@itemize

Itemize widens both the right and left margins, fills and justifies the text, and places a tick mark ("·") at the beginning of the first line of each new paragraph. Separate paragraphs as you type them with a blank line.

@enumerate

Enumerate is almost the same as @Itemize except that numbers instead of tick marks are placed before each paragraph.

@description

Description was used to format the text on this page. The word or phrase to be described is typed, followed by @\ (Scribe's tab indicator), followed by the description. Each word or phrase and its description are considered a paragraph, so separate each group with a blank line.

@multiple

This command allows multiple paragraphs within a single entry for itemize, enumerate, and description.

@blankspace[n lines] or [n inches]

This command tells Scribe to insert blank lines in your text. Use @blankspace[5 lines] to insert 5 blank lines. Use @blankspace[3.4 inches] to insert a vertical space that size.

@\ Use this to indicate a tab (instead of the tab key on your terminal). Be sure to use the \ (back slash) and not the / (forward slash). It is often necessary to use the Format or Verbatim command around the text with tabs so that the text is not filled and justified. (See the Scribe manual for examples of using tabs.)

@> This command pushes whatever text follows it to the right margin (flush right). When @> is used with a tab, the text will be flushed against the tab instead of the right margin.

@ <space>

Typing one @ sign followed by one space creates one and only one space in the text. Typing @ @ @ (three @ signs, each followed by a space) creates exactly three spaces in your text.

@@ Typing two @ signs together in the .mss file causes one @ sign to be printed in your text.

@newpage

This command causes a page break.

@* This command causes a line break. It is not recommended that you use this command often. (See warnings in Scribe manual!)

@group

Group prevents the text from being broken between pages. If you have a paragraph or chart that is being broken up between pages in an awkward way, this command will cause it to be printed on one page. Usually this should be used in the @begin[group] ... @end[group] command format.

20.4. SENDING AN *.MSS* FILE THROUGH SCRIBE

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

When your .MSS file is ready to be sent through Scribe, type the following from the EXEC level (underlined words are what you type):

@Scribe⇐

Scribe 4(1400) Copyright (C) 1981, 1984 UNILOGIC, Ltd.

Welcome to Scribe

*FileName.MSS⇐

[Processing *FILENAME.MSS.1*

[Device *IMPRINT10*

[RawFontDirectory *Imprint10*]

] Document type *Text*

[FontFamily ComputerModernRoman12]

]

[1 2 3]

4.

**PS:<USERNAME> *FILENAME.IMP* for device IMPRINT10 has 4 pages.

Ⓢ

In this example, Scribe processed the .MSS file, prepared a .IMP file for the Imagen, and returned to the EXEC level. The .IMP file can now be sent directly to the Imagen with the Laser command.

To process a manuscript file through Scribe for a device other than the one specified by the @device command in the file, when typing in the FileName, follow it by a forward slash and the name of the device you would like (for example, filename.MSS/lpt).

20.5. EXAMPLE OF .MSS AND .IMP FILE

Sample .MSS file (see next page for the .IMP version).

```
@device[IMPRINT10]
@style[indentation 0]
@majorheading[Fun with Scribe]
```

The purpose of this file is to demonstrate the results of using a few of Scribe's simple commands.

```
@i[Here is an example of the use of the italics command.]
@u[Here is an example of the use of the underline command], and
here is an example of the
use of the @center[center command.]
```

Look at this page to see the command or ".MSS" file, and look at the next page for the ".IMP" file (the file Scribe produced by following the commands in the .MSS file).

The "itemize" and "enumerate" commands are also very simple to use.

Use them with the @@Begin and @@End commands, and separate each paragraph with a blank line. The following was produced with the @@enumerate command:

```
@begin[enumerate]
```

The only difference between "itemize" and "enumerate" is that "enumerate" numbers each paragraph and "itemize" puts a tick mark by each paragraph.

It is very important to remember to "end" any command that was started with a "begin." If you do forget, there will be at least one error message when the file is run through Scribe.

```
@end[enumerate]
```

```
@blankspace[1 line]
```

```
@begin[description]
```

Example@\The description command separates a word or phrase from text. If the word or phrase overlaps the tab, the text will start on the next line. Separate paragraphs with a blank line.

```
@end[description]
```

```
@blankspace[1 line]
```

```
@begin[quotation]
```

The Quotation command causes both margins to be indented, fills and justifies the text, and inserts a blank space before and after the text delimited by the command. The Verse command is similar, but does not fill and justify.

```
@end[quotation]
```

.IMP file

Fun with Scribe

The purpose of this file is to demonstrate the results of using a few of Scribe's simple commands. *Here is an example of the use of the italics command.* Here is an example of the use of the underline command, and here is an example of the use of the

center command.

Look at this page to see the command or ".MSS" file, and look at the next page for the ".IMP" file (the file Scribe produced by following the commands in the .mss file).

The "itemize" and "enumerate" commands are also very simple to use. Use them with the @Begin and @End commands, and separate each paragraph with a blank line. The following was produced with the @enumerate command:

1. The only difference between "itemize" and "enumerate" is that "enumerate" numbers each paragraph and "itemize" puts a tick mark by each paragraph.
2. It is very important to remember to "end" any command that was started with a "begin." If you do forget, there will be at least one error message when the file is run through Scribe.

Example

The description command separates a word or phrase from text. If the word or phrase overlaps the tab, the text will start on the next line. Separate paragraphs with a blank line.

The Quotation command causes both margins to be indented, fills and justifies the text, and inserts a blank space before and after the text delimited by the command. The Verse command is similar, but does not fill and justify.

20.8. TABS

Sample .MSS file (see next page for .IMP version).

@subsection[Setting Tabs]

@begin[format]

@tabclear

@tabs[10,20,30,40]

@\344@\4859@\5938@\495

@\3988@\3975@\384@\394

@\@\@\397@\2987

@tabclear

@tabdivide[5]

@\234@\8796@\9876@\986

@\2368@\123@\6797@\23

@\37545@\2346@\3759@\234

@subsection[Tabs used with the Flushright Command]

@tabclear

@tabs[25,28]

@> Name: @\David Roode

@> Street Address: @\333 Ravenswood Avenue

@\Menlo Park, CA 94025

@> Work phone: @\ (415) 859-2025

@subsection[Setting Tabs Visually]

@tabclear

These people will arrive at 10:00 a.m.: @^Francine Perillo

@\Mary Stahl

@\Elizabeth Redfield

@end[format]

.IMP file

20.6.1. Setting Tabs

344	4859	5938	495	
3988	3975	384	394	
		397	2987	
	234	8796	9876	986
	2368	123	6797	23
	37545	2346	3759	234

20.6.2. Tabs used with the Flushright Command

Name: David Roode
 Street Address: 333 Ravenswood Avenue
 Menlo Park, CA 94025
 Work phone: (415) 859-2025

20.6.3. Setting Tabs Visually

These people will arrive at 10:00 a.m.: Francine Perillo
 Mary Stahl
 Elizabeth Redfield

20.7. USING SCRIBE FOR LETTERS

To create letters formatted for letterhead stationery, use `@Make[Letterhead]`. This will create a file with the SRI Letterhead.

Device options are:

- `@Device[IMPRINT10]` Use this for producing final copy
- `@Device[Lpt]` Use this for producing a preliminary hardcopy to proofread
- `@Device[File]` Use this for proofreading a letter online.

The command `@String[Name=" "]` (shown in the Scribe manual in the sample .MSS file) is useful only for letters with more than one page. After page one, the name given in the string (between the quotes) will appear at the top left of each succeeding page along with the page number in the center and the date on the right.

For paragraph indentation, use `@Style[Indentation 5]` (to indent 5 spaces) or `@Style[Indentation 0]` (for no indentation).

Imprint10 Letter

If you have created a .IMP file for the Imprint10, you will use the Laser command from the EXEC level to send the file to the Imagen. After the file has been printed on the Imagen, you can photocopy it onto letterhead stationery, if desired.

Sample .MSS File for Imagen letter (see next page for .IMP version).

@Make[Letterhead]
 @Device[IMPRINT10]
 @String[Name="Mr. Michael Jackson"]
 @Begin[Address]
 Mr. Michael Jackson
 111 Superstar Street
 Numberone, Everywhere 11111
 @End[Address]
 @Begin[Body]
 @Greeting[Dear Michael,]

Your last album, "Thriller," was truly thrilling and I loved every minute of it. "Beat It" was really spacy and all the tracks were special in some way. The problem is, I played it sooo much, I'm sick of it and now I can't wait for your next one. Based on past experience, I'm afraid that you will keep me in agony for another year and I just want you to know that I will come unglued if you do!

I also want to know when you're going to give another concert in Los Angeles. All the kids here are saving their money for that magic day when tickets for your concert go on sale and they can wait in line all night to be the first to get them.

Don't forget, if you ever need an enthusiastic president for your Michael Jackson Fan Club, you know you can call on me.

@End[Body]
 Yours forever,

Ms. Fan Ardent
 @Begin[Notations]
 FA/fa

cc: @^Mr. Paul McCartney
 @\Mr. Vincent Price

Enc. "A Groupie's View of Michael Jackson"
 @End[Notations]

.IMP file

January 26, 1984

Mr. Michael Jackson
111 Superstar Street
Numberone, Everywhere 11111

Dear Michael,

Your last album, "Thriller," was truly thrilling and I loved every minute of it. "Beat It" was really spacy and all the tracks were special in some way. The problem is, I played it sooo much, I'm sick of it and now I can't wait for your next one. Based on past experience, I'm afraid that you will keep me in agony for another year and I just want you to know that I will come unglued if you do!

I also want to know when you're going to give another concert in Los Angeles. All the kids here are saving their money for that magic day when tickets for your concert go on sale and they can wait in line all night to be the first to get them.

Don't forget, if you ever need an enthusiastic president for your Michael Jackson Fan Club, you know you can call on me.

Yours forever,

Ms. Fan Ardent

FA/fa

cc: Mr. Paul McCartney
Mr. Vincent Price

Enc. "A Groupie's View of Michael Jackson"

21. USEFUL PROGRAMS

Key:
 ↑ = Control Key
 ≈ = Escape Key
 ⇐ = Carriage Return

The programs listed below are currently available on all SRI TOPS-20 systems. They are presented alphabetically, with a brief description of their functions followed by online filenames which contain more information.

COOKIE

This is a whimsical program useful when a user has had "one of those days." Typing `cookie⇐` at the EXEC level will produce a one-line message of profound wisdom, the kind often found in fortune cookies. For example, "The plural of spouse is spice."

DELVER

This program will delete lower versions of a specified file. To run the program, type at the EXEC: `delver⇐`. The program will ask the user if the lowest and next-to-highest versions should be retained, then prompt for a filename. More information can be found in `help delver⇐`

FINGER

This command will print an alphabetical list of everyone who is currently logged in on the system, along with each user's personal name, his/her job number, the number of the terminal s/he is currently logged in on, and the name of the program he is using at that moment. Information about a specific individual user can be obtained by including that user's directory name after the Finger command, i.e., `Finger DirectoryName⇐`. More detailed information can be found in `help finger⇐`

KERMIT

KERMIT is a protocol for transferring sequential files between computers of all sizes over ordinary asynchronous telecommunication lines using packets, checksums, and retransmission to promote data integrity. A complete description of this program can be found in the files DOC:KERMIT.DOC and `help kermit⇐`

MPW

MPW is a program that displays a selection of suggested passwords. Every time it is started a new set of suggestions will be printed. To run the program, type from the EXEC: `mpw⇐`

PHOTO

PHOTO is a program to save transcripts of sessions with the computer. It is used most often on video terminals when you want a hard-copy record of your session. For more information, see `HELP PHOTO`.

REMIND

The REMIND program allows you to create reminders for yourself that are sent off at various intervals of your own choosing (daily, weekly, by the minute, etc.) by mail and/or a direct message to your terminal if you are logged in at the time. See `help remind⇐` and `DOC:REMIND.DOC`.

SPELL

This program is a spelling checker and corrector that is easy to use. Type `help spell⇐` or see `DOC:SPELL.DOC` for instructions on using this valuable program.

SORT

SORT is a high performance sort/merge package from Digital Equipment Corporation. It may be run as a stand-alone sort/merge, or embedded in a COBOL or FORTRAN program. For a summary of commands, type help sort← or see the file DOC:SORT.DOC for a detailed description of the program.

See also help ssort← for information about other sorting programs.

SPELL

SPELL is a program that checks text files for correctness of word spelling. In addition to the spelling check, SPELL provides a facility for correcting words that it thinks are misspelled. See help spell← for instructions.

WHOIS

The WHOIS program is used to display current information about individuals on the ARPANET and MILNET systems. The program accesses the network-wide database maintained here at the NIC. It is invoked by typing whois← at the EXEC level. To find out more about this very useful program, type help whois←

22. PROGRAMMING LANGUAGES

TOPS-20 is a powerful system for program development. Listed below are some of the currently available programming languages, followed by the names of files which contain online documentation.

BASIC

Beginner's All-purpose Symbolic Instruction Code. For help, type help basic←.

BLIS10

A dialect of BLISS, a language designed for writing large software systems. Type help blis10← for more information.

CC A compiler for the C language. Type help cc for more information.

FAIL

A fast, one-pass assembler for PDP-10 or PDP-6 machine language. Type help fail← for a brief introduction. A detailed description is in the file DOC:FAIL.MANUAL.

MACRO

Standard DEC assembler. Type help macro← or see the file DOC:MACRO.DOC for more information.

MIDAS

A PDP-10 assembler with macro and extensive string processing capabilities.

PASCAL

A high-level algorithmic language. Type help pascal← or see the files DOC:PASCAL.DOC and DOC:PASCAL.CHANGES for details.

SAIL

Stanford Artificial Intelligence Lab's ALGOL-like language. More information is contained in the files DOC:SAIL.DOC and DOC:SAIL.UPDATE. See also help sail←.

23. APPENDIX

1. TOPS-20 Commands with Brief Descriptions

<u>TOPS-20 Command</u>	<u>Description</u>
ACCESS	Grants owner and group privileges to directories and their files without changing your connected directory
APPEND	Joins two files together
ARCHIVE	Makes permanent off-line copies of files
ASSIGN	Allocates a device (usually a tape drive)
ATTACH	Attaches a job to your terminal
BREAK	Ends links made by TALK command
BUILD	Creates, modifies, or deletes a subdirectory
↑C	Interrupts current program, returns to EXEC
CANCEL	Withdraws queue requests from the batch, lineprinter, Imagen, or retrieval queues
CLOSE	Closes open files, releases JFNs
COMPILE	Translates source programs into object programs
CONNECT	Connects your job to a directory
CONTINUE	Continues a halted program
COPY	Makes copy of a specified file
CREATE	Creates a file
CREF	Translates .CRF files into listings
DAYTIME	Displays the date and time
DDT	Starts a debugging program
DEASSIGN	Gives up previously-assigned device
DEBUG	Debugs a program
DEFINE	Establishes or withdraws logical names
DEL Key	Deletes character to the left of cursor
DELETE	Marks files to be erased later by the <i>logout</i> or <i>expunge</i> command
DEPOSIT	Changes contents of a memory location

DETACH	Disengages a job from your terminal
DIRECTORY	Displays a list of filenames in a directory
DIRECTORY,	Allows you to enter subcommand mode for more directory information
DISABLE	Deactivates privileged capabilities
DISCARD	Gives up the tape copy of online files
DISMOUNT	Gives up access to a disk structure or tape set
EDIT	Edits an existing file
ENABLE	Activates privileged capabilities
END-ACCESS	Terminates ownership rights to a directory
EOF	Writes an end-of-file mark on a magnetic tape
EXAMINE	Inspects a memory location
EXECUTE	Compiles, loads, and starts a program
EXPUNGE	Erases deleted files from a directory
FDIRECTORY	Lists all information on files in a compressed format without headings
FINGER	Provides information on every user who is logged in on connected system
FINGER <i>USER</i>	Provides information on specific user
FORK	Specifies which process is current
GET	Places an executable program in memory
HELP	Displays helpful documentation on various system features
HELP ?	Provides a list of available help topics
HELP <i>TOPIC</i>	Displays online documentation for a specific command or program
INFO DISK	Displays information about current page allocation for directories and system
INFO FORK	Provides list of current forks
INFO JOB	Gives information about job status
INFO OUTPUT	Displays lineprinter and Imagen queues
KEEP	Retains fork for an interrupted program
LASER	Prints a hard copy of a file on the Imagen Imprint10

LIST	Provides hard copy of file from NIC lineprinter
LOAD	Compiles and loads a program in memory but doesn't start it
LOGIN	Begins a job
LOGOUT	Ends a job
MERGE	Merges an executable program with current memory
MODIFY	Changes output or batch requests
MOUNT	Mounts a disk structure or tape set
PLOT	Prints files on a plotter (The NIC does not currently support a plotter)
POP	Returns to superior level
PUSH	Begins an inferior level
↑Q	Resumes typeout on screen that was stopped by ↑S
QDIRECTORY	Displays a list of any deleted files in connected directory
R	Runs a system program
↑R	Redisplays current line
RDIRECTORY	Lists filenames with the date each file was last read
RECEIVE	Allows others to talk (link) with your terminal
REENTER	Starts the program in memory at restart address
REFUSE	Disallows system messages and disallows others to talk (link) with your terminal
REMARK	Treats subsequent input as comment only (until ↑Z is typed)
RENAME	Changes a filename
RESET	Terminates current process and clears its memory
RETRIEVE	Retrieves off-line files to disk
REWIND	Rewinds a magnetic tape or tape set to load point
RUN	Runs an executable program
↑S	Freezes typeout on screen
SAVE	Stores copy of memory in a file
SET	Sets various job parameters

SET ACCOUNT	Changes job account
SET DIR PASS	Changes password
SET FILE ACC	Assigns file account
SET FILE PROT	Sets file protection
SKIP	Moves a magnetic tape or tape set forward
START	Starts program in memory at start address
START N	Starts a program at a specific location
SUBMIT	Submits control files to the batch system
SYSTAT	Gives information about system and job status
SY USER	Provides information and job number for specified user
TAKE	Starts processing of a command file
TALK	Creates a communication link with another user
TDIRECTORY	Lists filenames with date and time each was last changed
TERMINAL	Sets various terminal options
TERMINAL ?	Lists options for defining terminal type
TRANSLATE	Gives directory names for ppn's
TYPE	Prints files on your terminal
↑U	Deletes line
UNATTACH	Disengages another job from its terminal
UNDELETE	Restores deleted files
UNLOAD	Unloads a magnetic tape
VDIRECTORY	Lists filenames with protection, size, date and time of last change, and last writer to each file
↑W	Deletes word to the left of current cursor position
WDIRECTORY	Lists filenames with write dates, creator and last writer to each file
YDIRECTORY	Lists filenames last written by the user in the connected directory

Useful Addresses

Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02238

Digital Equipment Corporation
P.O. Box CS-2008
Nashua, New Hampshire 03061

Document Distribution
USC/Information Sciences Institute
4676 Admiralty Way, Suite 1100
Marina del Rey, California 90291

Massachusetts Institute of Technology
Artificial Intelligence Laboratory
Attn: Publications
545 Technology Square
Cambridge, Massachusetts 02139

SRI International
Network Information Center
333 Ravenswood Avenue
Menlo Park, California 94025

Unilogic, Ltd.
160 N. Craig
Pittsburgh, Pennsylvania 15213

Suggested Reading

- [ARPA 82] *ARPANET User's Guide*, Computing Analysis Corporation,
Arlington, Va., 1982
- [DEC 80a] *TOPS-20 User's Guide*, Digital Equipment Corporation,
Marlboro, Mass., 1980
- [DEC 80b] *Getting Started with Decsystem-20*, Digital Equipment Corporation,
Marlboro, Mass., 1980
- [DEC 80c] *TOPS-20 Commands Reference Manual*, Digital Equipment Corporation,
Marlboro, Mass., 1980
- [LOTS 84] *LOTS Overview Manual*, Stanford University Bookstore.
- [Unilogic 84] Unilogic, *Scribe Document Production System User Manual*, Third Edition,
Unilogic, Ltd., Pittsburgh, Pa., 1980
- [Stallman 80] Stallman, Richard M., *EMACS Manual for TWENEX Users*,
Massachusetts Institute of Technology, AI Memo 555, September 1980.
Corresponds to EMACS Version 150.

